Scalable Visual Pretraining for Language Intelligence

Yiming Zhang[1,2,*], **Zhonghan Zhao**[1,3,*], **Wenwei Zhang**[1,*], **Haiteng Zhao**[1], **Tianyang Lin**[1], **Yunhua Zhou**[1], **Demin Song**[1], **Kuikun Liu**[1], **Haochen Ye**[1], **Haian Huang**[1], **Yuzhe Gu**[1,4], **Haijun Lv**[1], **Qipeng Guo**[1], **Bin Liu**[2], **Gaoang Wang**[3,†], **Kai Chen**[1,†]

[1]Shanghai Artificial Intelligence Laboratory [2]University of Science and Technology of China
[3]Zhejiang University [4]Shanghai Jiao Tong University
*These authors contributed equally to this work. † Corresponding authors.
gaoangwang@zju.edu.cn, chenkai@pjlab.org.cn

The rapid progress of large foundation models has been driven predominantly by pretraining on large-scale text corpora. However, many forms of knowledge are conveyed through visual representations, where figures, typeset equations, and page layouts carry rich information that cannot be faithfully or completely captured by text alone. Yet current pretraining approaches discard these visual cues by converting visually rich sources, such as documents and web pages, into plain text for learning language intelligence. This paper challenges the default assumption that language models must be trained on text-only representations and shows that Visual Pretraining is a scalable learner for foundation model intelligence. To this end, we conduct a systematic study of unsupervised visual pretraining paradigms that directly leverage visual documents without text extraction. Across multiple backbones and benchmarks, visual pretraining on the same underlying corpora consistently outperforms text-only pretraining, offering an efficient pathway to scalable language intelligence.

1. Introduction

The striking advances of large foundation models have been driven by pretraining on text corpora at unprecedented scale [4, 9, 12]. While effective, this paradigm rests on a strong implicit assumption: that all knowledge worth learning can be losslessly encoded as a linear sequence of text tokens. However, cognitive science has long shown that humans routinely reason with diagrams, spatial layouts, mathematical notation, and other representational forms whose visual cues make certain relations directly available for inference [1, 2, 15, 38]. Converting these visual cues into plain text before training is therefore inherently lossy. Here we challenge this assumption and show that foundation models can learn directly from visual corpora without text extraction or image-text pairing supervision, yielding stronger language intelligence than text pretraining on the same underlying corpus.

Scientific documents are a particularly acute case of this loss. Papers, textbooks, and technical reports communicate complex content through figures, tables, formula layouts, and page-level spatial organization, all of which encode geometric constraints, symbolic topologies, and structural correspondences essential to scientific reasoning. The recently proposed Platonic Representation Hypothesis [10] formalizes a related intuition for machine learning, arguing that representations learned by different models and modalities converge toward shared abstractions of reality. These observations imply that the linguistic information extracted from a scientific document is only a projection of a richer underlying structure, and that this structure could in principle be learned directly from raw visual documents.

Yet existing approaches do not exploit this possibility along either of two dimensions. At the data level, pretraining corpora reduce visual documents to plain text, either by parsing HTML or LaTeX source [7, 27] or by applying neural document-parsing models to PDFs as a preprocessing step [3, 13, 16, 32], after which the language model is trained exclusively on the resulting text. At the training level, recent multimodal foundation models do incorporate visual modalities during training [5, 21, 33], but they treat visual inputs as conditioning context for text-token prediction, preventing visual content from being deeply integrated into the predictive

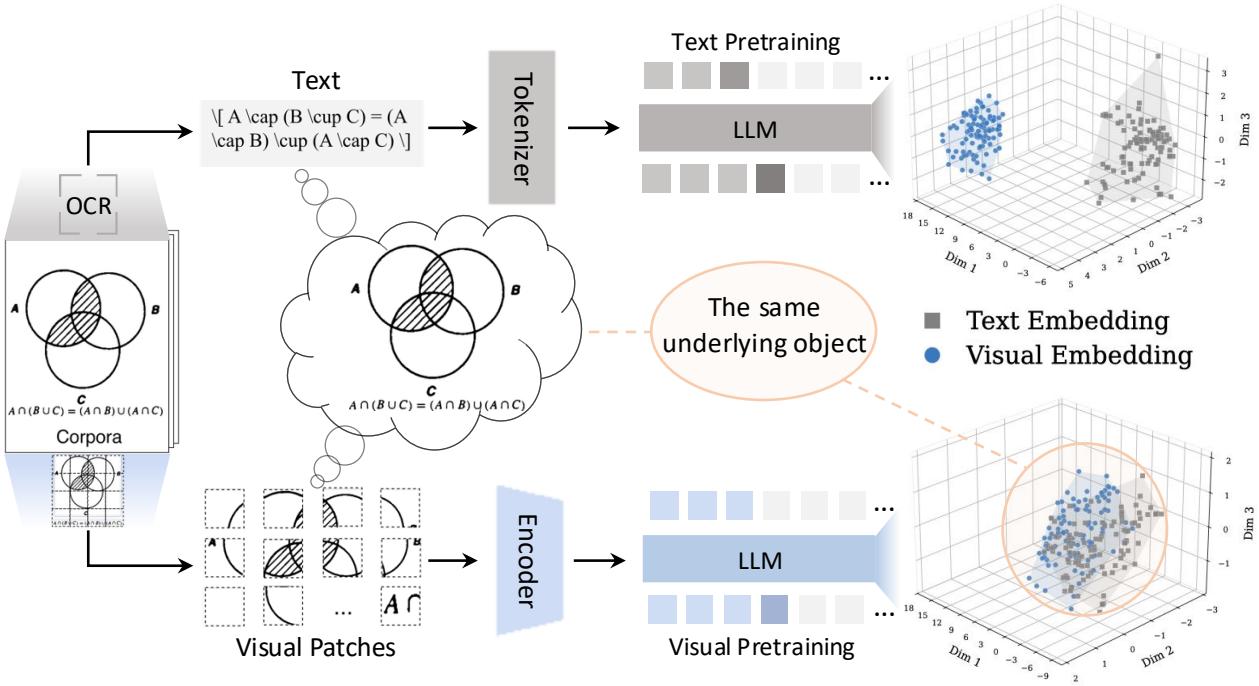

Figure 1: **Matched text and visual pretraining from the same scientific-document corpus.** In the TP pathway, PDF pages are parsed into text, tokenized, and trained with text-token prediction, which can discard or distort page-level structure such as diagrams, equations, tables, and layout. In the VP pathway, the same pages are rendered as images, filtered into foreground visual patches, and trained with next visual latent prediction in a frozen visual-feature space. The t-SNE [31] projections on the right illustrate the representation change quantified in Table 2: VP brings matched visual and textual document embeddings closer in the shared representation space.

process. In both regimes the raw documents are consumed and then discarded leaving the visual modality outside the model's learning objective.

Here we present Visual Pretraining (VP), a framework in which a foundation model learns visual information directly from raw documents without any text extraction or image-text pairing supervision. Through a systematic study under carefully matched data sources, we find that across multiple LLM backbones and scientific reasoning benchmarks, VP consistently outperforms text-only pretraining on the same underlying corpus, is substantially more efficient (using only 25% of the token budget) with respect to both model size and data scale, and strengthens cross-modal alignment, establishing VP as a scalable pathway for learning both language and visual intelligence. This work makes three contributions. First, we show that reducing visual documents to plain text incurs substantial information loss, and that visual pretraining recovers this otherwise discarded information. Second, we introduce an autoregressive visual pretraining framework that trains a foundation model to predict document patches in latent space, deeply integrating visual latent into the predictive process. Third, through a unified empirical study with matched corpora across multiple architectures and benchmarks, we establish VP as an effective, efficient, and scalable alternative to text-only pretraining.

2. Results

Our experiments establish three findings. First, visual pretraining (VP) outperforms text pretraining (TP) on scientific reasoning under matched corpora. Second, this gain scales efficiently with training data. On the same underlying document corpus, VP surpasses TP while consuming only 25% of the token budget. Third, without any image–text pair supervision, VP improves visual perception as well as reasoning across modalities. Throughout the experiments, VP and TP use the same text corpus and a matched additional scientific-PDF corpus in the continued pretraining (CPT) stage. VP trains on raw document pages represented as visual tokens, whereas TP trains on parsed text from the same documents. This setup isolates the effect of preserving the

Table 1: VP improves text-only scientific reasoning under matched document sources. All rows use the same starting checkpoint and SFT stage. TP and VP differ only in how the scientific-PDF corpus is represented: MinerU2.5-parsed text for TP and rendered page images for VP. Scores are reported on text-only reasoning benchmarks; GPQA denotes GPQA Diamond and AIME denotes AIME 2025.

Method	Multimodal Models				Language Models			
	MMLU-Pro	GPQA	AIME	HLE	MMLU-Pro	GPQA	AIME	HLE
	Qwen 3.5 [29]				*Qwen 3* [36]			
Base	82.31	77.84	81.56	14.39	81.21	75.06	75.44	10.59
Text Pretraining	83.91	76.24	89.58	15.70	81.52	74.94	74.99	11.39
VP (Ours)	85.09	79.29	90.21	16.67	81.94	77.08	76.98	11.77
	Llama 3.2 Vision [25]				*Llama 3.1* [8]			
Base	47.24	27.46	8.02	6.41	59.48	39.71	24.17	6.75
Text Pretraining	50.60	30.24	13.75	6.08	60.64	43.88	23.65	6.79
VP (Ours)	51.52	33.08	18.54	7.00	62.77	47.10	24.27	7.17

native visual form of scientific documents. Then we apply identical supervised fine-tuning (SFT) to all VP and TP checkpoints before evaluation to elicit the structured reasoning traces and answer formats expected by the benchmarks.

Effectiveness: VP improves scientific language reasoning We first establish that continued pretraining on raw document pages yields stronger performance on scientific reasoning benchmarks than continued pretraining on textualized documents from the same corpus. We compare the base model, the matched TP baseline, and VP under identical document sources and SFT data. The two pretraining settings differ only in how documents are represented. Experiments are conducted across state-of-the-art foundation models, including Qwen3.5 [29], Qwen3 [36], Llama3.2 Vision [25] and Llama3.1 [8]. Unless otherwise specified, we report average pass@8 for GPQA [30], the average score over 32 runs for AIME-25 [24], and pass@1 for MMLU-Pro [34] and HLE [28]. Results are reported in Table 1.

VP consistently improves over the matched TP baseline, and the gains appear across both native multimodal models and language-only models, indicating that the benefit is not tied to a single architecture. Because the two settings differ only in document representation, the improvement on reasoning benchmarks suggests that VP is an effective way to acquire language intelligence directly from visual corpora, which preserve higher-fidelity reasoning-relevant information that text extraction weakens or discards. Specifically, the consistent improvements on multi-disciplinary benchmarks indicate that VP successfully internalizes knowledge from the visual content of scientific documents. GPQA Diamond improves by up to 3.22 points across the four backbones (e.g., 76.24 to 79.29 on Qwen 3.5) and MMLU-Pro by up to 2.1 points, consistent with the view that source-document knowledge in these scientific domains is partly carried by geometric figures, physics schematics and other visual content that text extraction cannot faithfully preserve. HLE, by contrast, shows only marginal improvements (up to 0.97 points), since its difficulty derives primarily from hard multi-step reasoning rather than the knowledge gained from visual information.

Scalability: gains scale with VP We further demonstrate that the benefit of VP is efficient and scalable. Figure 2(a) compares the training loss dynamics of VP and TP across CPT and SFT stages. During CPT, the two curves are close, with VP reaching a slightly lower final loss. After both checkpoints are fine-tuned with the same SFT data, the VP checkpoint converges faster and reaches a lower final SFT loss. We attribute this discrepancy to the limited sensitivity of pretraining loss to downstream capabilities [35]. SFT loss, by contrast, is grounded in the geometric structure established during pretraining [17], and thus more directly reflects representation quality and its capacity for reasoning.

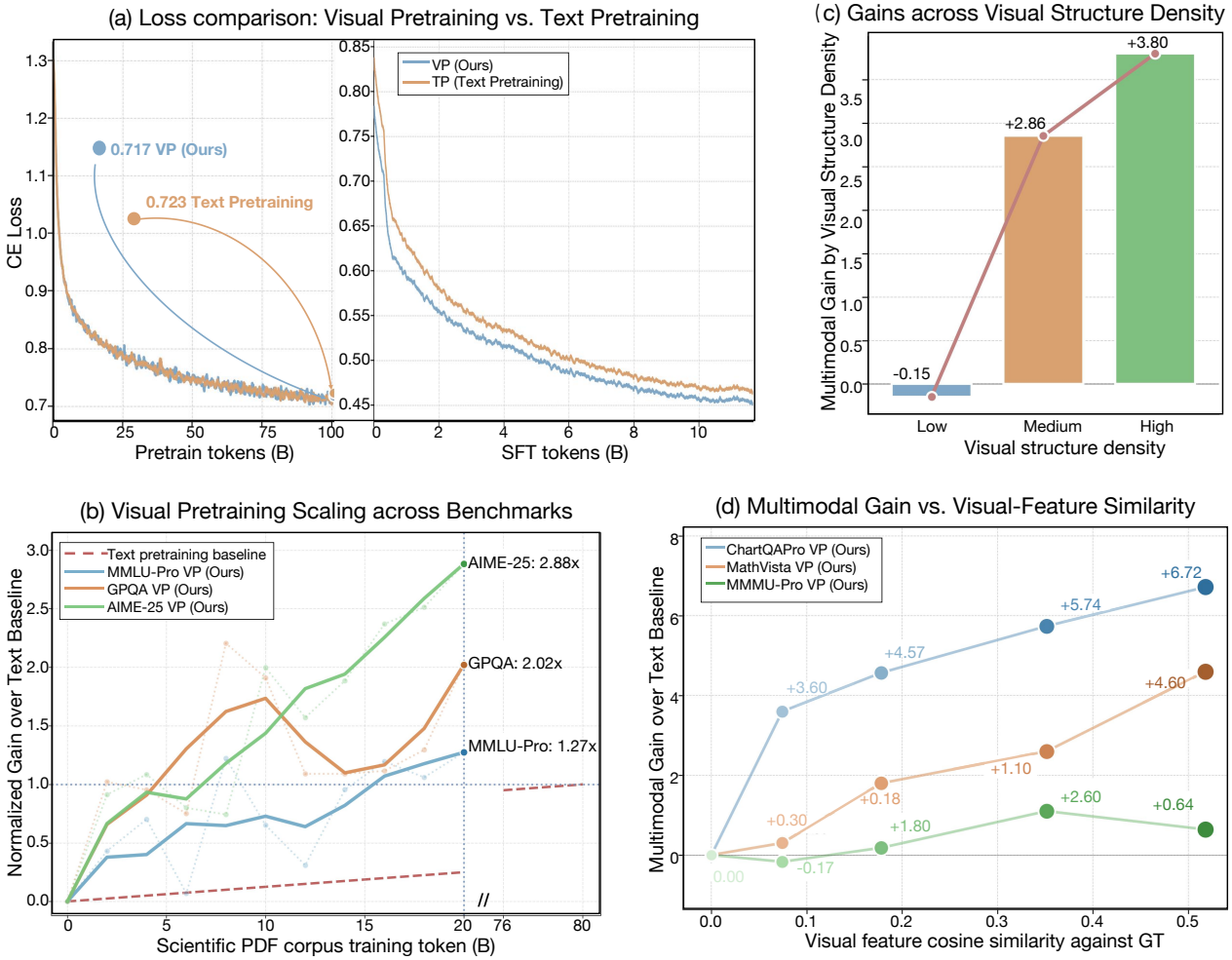

Figure 2: Visual pretraining scales with retained PDF visual tokens and benefits most from structure-heavy pages. (a), VP reaches a more favorable SFT trajectory after comparable CPT loss. **(b)**, Increasing the retained visual-token budget consistently raises downstream gains, normalized by the TP-over-base improvement. **(c)**, VP's advantage is largest on examples with high visual-structure density, where figures, equations, tables, and layout carry more of the evidence. **(d)**, Better next-token visual-feature prediction is associated with larger multimodal gains over TP, linking visual-space modeling quality to downstream transfer.

We then examine how downstream performance scales with training dynamics. In Figure 2(b), we normalize the downstream gain of VP against that of TP, both measured relative to the starting checkpoint. For the same PDF corpus, VP trains on only approximately 20B visual tokens, while TP processes roughly 80B text tokens. Moreover, VP achieves steadily increasing normalized gains as training progresses, reaching 1.27× on MMLU-Pro, 2.02× on GPQA and 2.88× on AIME-25, indicating superior scaling efficiency. Figure 2(d) further reveals that visual feature cosine similarity, a proxy for visual prediction quality, correlates strongly with downstream improvements, suggesting that better visual space modeling supports stronger reasoning, by enabling richer knowledge extraction from visual content. To verify that these gains are visually grounded, we categorize the evaluation sets by visual-structure density following Masry et al. [23] (Figure 2(c)). The advantage of VP over TP grows with density. On low-density, text-dominant pages, VP and TP perform on par, whereas high-density pages rich in figures, equations, and tables produce substantially larger improvements. This confirms that VP's scaling benefits are genuinely driven by its modeling of visual content.

Efficiency: compact visual representations preserve VP gains We next demonstrate the efficiency of VP. First, visual representations compress the same scientific documents far more effectively than OCR-extracted text. Specifically, VP trains on approximately 20B visual tokens, whereas TP consumes roughly 80B parsed

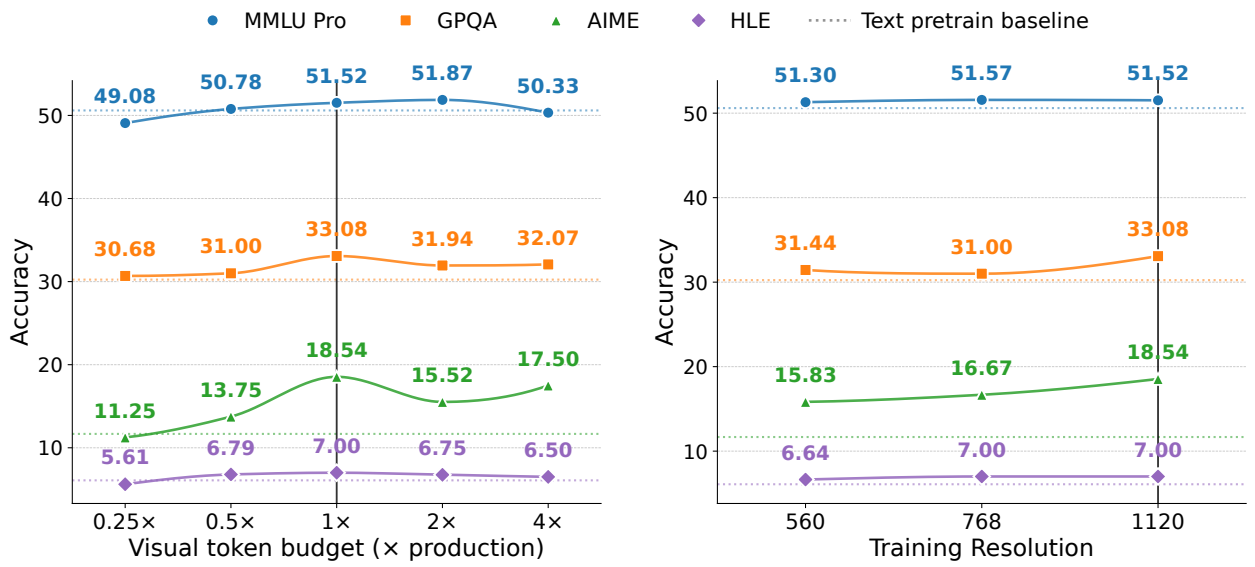

Figure 3: **The main VP setting balances visual-token budget and rendering resolution. Left**, with the training context fixed for each forward pass, varying the visual token budget shows that the 1× setting (8,192 foreground visual tokens per batch) offers the best trade-off. **Right**, with the visual budget fixed at 8,192 tokens, lowering the maximum resolution preserves strong performance, indicating the efficiency of VP under compressed visual input.

text tokens. Second, under a fixed optimization token budget, VP achieves strong gains with only a modest share of visual tokens. We vary the visual token budget within a fixed training context (Figure 3, left). The 8,192-token setting (1×) achieves the best trade-off. We hypothesize that the suboptimal performance at larger visual-token budgets stems from optimization effects in the visual InfoNCE training. When more visual tokens are retained, the loss is computed over more prediction targets and in-batch contrastive terms, which can change the effective gradient scale and optimization behavior. Moreover, because we keep the learning rate and other optimization hyperparameters fixed in this sweep, larger token budgets may not be fully tuned under this training recipe. Finally, the visual input itself admits further compression. We train VP at varying resolutions and evaluate downstream performance. With the visual budget fixed at 8,192 tokens, lowering the maximum resolution preserves strong performance (Figure 3, right), and VP still outperforms TP. This further amplifies the efficiency of VP.

Cross-modality: multimodal capabilities emerge without paired supervision We finally demonstrate that the gains from VP generalize beyond text-only reasoning. Although VP is performed on unlabeled raw document pages, without task-specific multimodal annotations, it also improves multimodal representation and downstream visual reasoning.

We extract hidden states from the foundation model (*i.e.*, Qwen3.5) on 100 image–text pairs sourced from scientific documents, and compare the pooled visual and text embeddings before and after VP. Table 2 (a) shows consistent gains across three complementary levels of alignment. Globally, the centroid separation [19] drops from 1.665 to 0.661 and the paired cosine similarity rises from 0.631 to 0.907, indicating that the two modalities now share a common subspace and that paired samples are systematically co-located. Structurally, Linear CKA [14] improves from 0.657 to 0.745, evidencing agreement on the larger geometric structure of the two representation spaces. Locally, Mutual k-NN overlap [10] improves at all evaluated k, confirming neighbourhood alignment at the instance level. We also provide a qualitative view in Figure 1, where the originally disjoint visual and text embedding clusters collapse into a single overlapping region after VP pretraining. For visual comparability and fairness, both panels share anchor features and t-SNE settings.

Table 2 (b) reports multimodal benchmark (*i.e.*, MMMU-Pro [37], SFE [39], ChartQAPro [23] and MathVista [22]) performance with Qwen 3.5 and Llama 3.2 Vision. VP consistently outperforms both the pretrained base and the text-pretrained model on every benchmark and backbone. TP yields negligible gains and occasionally

Table 2: VP improves cross-modal alignment and multimodal transfer without labeled multimodal pretraining data. Left **(a)**: alignment metrics on 100 held-out scientific document image–text pairs before and after VP; lower centroid separation and higher cosine similarity, CKA, and mutual k-NN indicate better alignment. Right **(b)**: pass@1 performance on multimodal benchmarks for native multimodal models, comparing the base checkpoint, TP, and VP. The gains show that unlabeled visual-document pretraining improves both representation compatibility and downstream visual reasoning.

(a) **Cross-modal alignment.**

Metric	Original	VP (Ours)	Δ
	Global		
Centroid Sep. ↓	1.665	0.661	−1.004
Cosine Sim. ↑	0.631	0.907	+0.276
	Structural		
Linear CKA ↑	0.657	0.745	+0.088
	Local		
Mutual k-NN@1 ↑	0.140	0.310	+0.170
Mutual k-NN@5 ↑	0.288	0.420	+0.132
Mutual k-NN@10 ↑	0.395	0.496	+0.101

(b) **Multimodal benchmark performance.**

Method	MMMU-Pro	SFE	ChartQAPro	MathVista
	Qwen 3.5 [29]			
Base	71.39	53.41	57.99	84.30
Text Pretraining	72.14	53.09	56.42	85.50
VP (Ours)	73.87	56.57	61.80	86.70
	Llama 3.2 Vision [25]			
Base	28.55	28.86	20.95	39.80
Text Pretraining	28.21	29.36	22.23	39.60
VP (Ours)	29.19	31.08	27.67	44.40

regresses (*e.g.*, $57.99 \rightarrow 56.42$ on ChartQAPro for Qwen 3.5), indicating that further TP cannot harvest information from visually rich documents. The VP advantage over TP is largest on visually heavy benchmarks, reaching $+5.4$ on ChartQAPro for both backbones and $+4.8$ on MathVista for Llama 3.2 Vision. Together with the density-stratified results in Figure 2 (c), this indicates that VP's main benefit lies in visual content (figures, equations, tables, and complex layouts) that text parsing cannot faithfully recover.

Reasoning cues in visual pretraining To understand the mechanism behind VP's scientific reasoning gains, we conduct a qualitative study of attention patterns. We compare attention maps elicited by the same math problem under text and visual inputs, using Qwen3.5 as the native multimodal backbone. In the textual setting, the problem and its chain-of-thought solution are provided as plain text; we extract attention from the last Transformer layer, average over heads, and use the final answer sentence as the query. In the visual setting, the same problem is rendered as an image and fed through the visual pathway. We use visual tokens in the answer region as queries and average their attention across retained patches. We show a representative example where both attention maps highlight corresponding semantic regions, including the question constraint and intermediate computation steps, while de-emphasizing filler content (Figure 4). This probe does not imply that the two modalities use identical internal mechanisms, but it shows that VP exposes reasoning-relevant visual evidence to the shared LLM in a way that parallels the evidence used in text-based reasoning.

3. Discussion

Large language models typically acquire language intelligence from text-only corpora, yet much scientific knowledge is originally expressed in multimodal documents. Scientific papers contain equations, figures, tables and carefully designed layouts, all of which can be weakened or distorted when converted into plain text. We asked whether language models can acquire stronger scientific reasoning by directly adding visual pretraining on the native form of such documents to standard text continued pretraining. Our results provide a cautiously affirmative answer, showing that VP improves scientific reasoning over matched text pretraining, scales efficiently with compact visual budgets, and transfers to multimodal reasoning without labeled cross-modal data. Together, these results show that raw scientific pages can provide unlabeled visual supervision that complements continued pretraining for foundation models.

The benefit of VP appears to come from preserving and modeling structure that is lost during textualization. Scientific pages are not arbitrary images, as their visual organization encodes equation topology, table structure, figure-text correspondence, symbolic continuity and spatial grouping. VP trains the shared autoregressive backbone on foreground visual tokens ordered by their page positions, using a causal objective that predicts

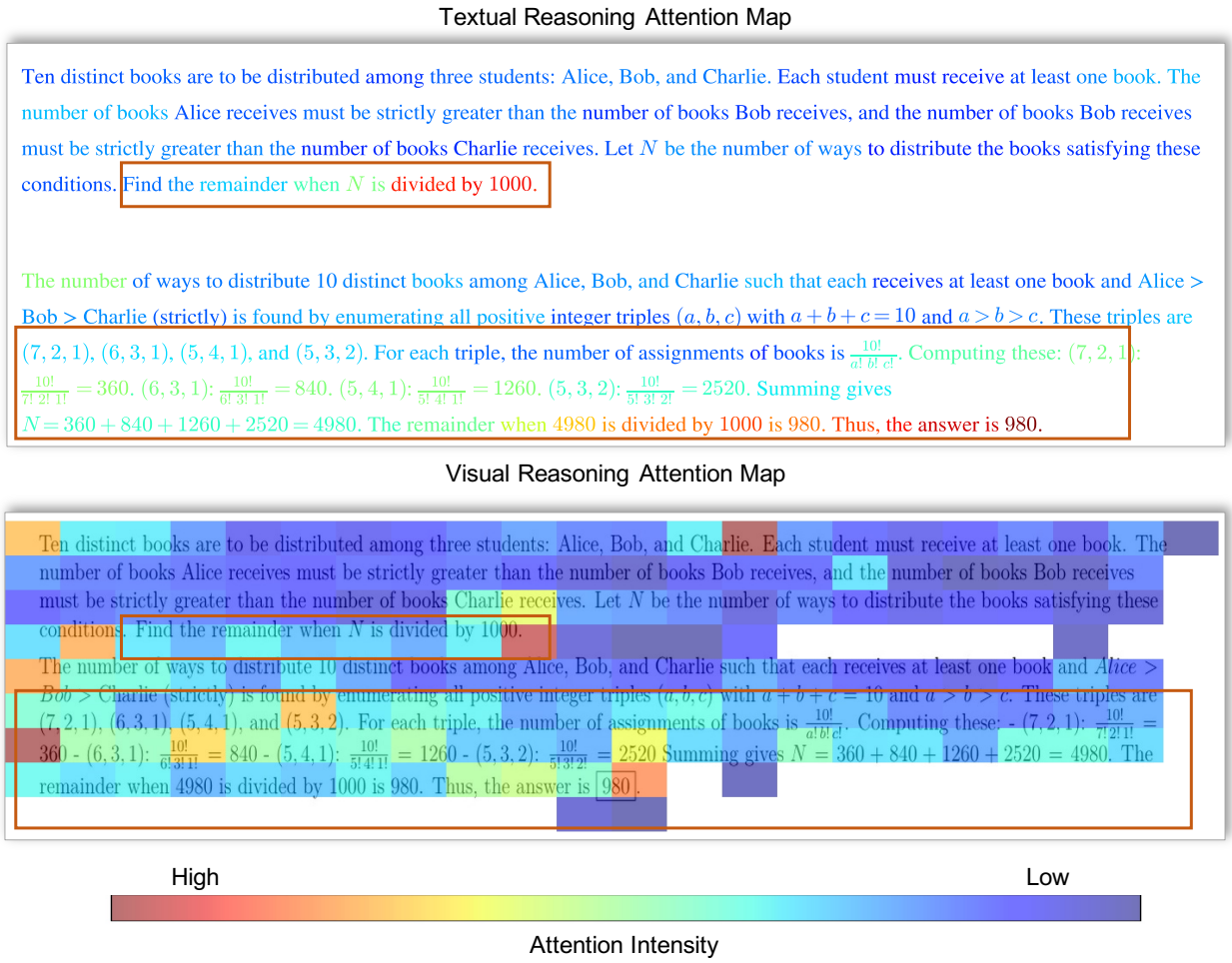

Figure 4: **Visual-token reasoning attends to the same semantic evidence as text reasoning.** We compare foundation models with TP or VP on the same math problem under text and image inputs. **Top**, textual reasoning: the problem and chain-of-thought solution are given as plain text, and attention is measured from the final answer sentence to previous tokens. **Bottom**, visual reasoning: the same problem is rendered as an image and fed through the visual pathway, and attention is measured from visual tokens in the answer region to retained page patches. Both views concentrate on the semantic regions highlighted in red, including the question constraint and intermediate computation steps.

the next visual latent. By making the visual document itself the prediction target, this objective encourages the backbone to model visual representations in a way that preserves document-native relations. Several observations support this interpretation. First, VP improves scientific reasoning over matched text pretraining across backbones, suggesting that the model can acquire reasoning-relevant knowledge from visual elements such as equations, figures, tables and layouts. Second, in multimodal evaluations, the gains are largest on visually dense examples, where figures, equations, tables and layout carry more of the evidence needed for reasoning. Cross-modal alignment also improves without paired image-text supervision, suggesting that the visual objective reshapes the shared representation space in a way that becomes more compatible with language. Finally, the qualitative attention analysis suggests that visual-token reasoning can attend to semantically relevant regions of the rendered problem, although this should be read as supporting evidence rather than a proof that text and visual reasoning use identical mechanisms.

Text continued pretraining remains an effective way to absorb knowledge that is already well represented in language, but scientific documents reach the model through a textualization pipeline that can discard layout, formulas, diagrams and other visual relations. VP does not replace this pathway. Instead, it mixes standard text continued pretraining with a visual next-latent objective on native scientific pages, allowing the model to retain

text exposure while learning from the native visual form of the same scientific-document corpus. Because VP and the text baseline use the same document source, the gain suggests that the native visual representation provides useful supervision beyond the textualized content alone.

Existing multimodal pretraining methods usually introduce visual inputs through image-text pairs, captioning losses, contrastive alignment, OCR supervision or document-understanding objectives. OCR-free document models such as Donut [13] and Pix2Struct [16] further show that document images can be used directly, but their pretraining is mainly optimized for parsing, understanding or image-to-text generation. In contrast, VP treats rendered scientific pages as unlabeled visual sequences and trains the shared autoregressive backbone with a next-latent prediction objective. It does so without dense annotations or explicit image-text pairing.

Our study has several limitations. First, our approach should not be interpreted as a form of visual pretraining that is fully independent of language pretraining. Since VP is grounded in text pretraining and introduces visual-latent prediction during continued pretraining, the observed gains should be understood as evidence that visual pretraining can complement and extend foundation model pretraining. Second, while VP shows that visual-latent prediction can effectively guide pretraining optimization, future work may further investigate how to better coordinate visual-latent decoding with text decoding. Our observations suggest that visual pretraining strongly encourages the model to capture visual presentations and structures, motivating future studies on loss scheduling, foreground-aware token selection, and parameter efficient visual modules. Third, our experiments primarily focus on high knowledge-density scientific PDFs, where visual layouts, figures, tables, formulas, and surrounding textual context are tightly coupled to semantic meaning. It remains an open question whether the same pretraining strategy transfers to broader visual corpora such as natural images or video.

This work points to two promising directions. First, visual pretraining provides a new route for foundation model training. For high-density visually native corpora, such as scientific documents, charts, tables, formulas, and structured layouts, visual-latent prediction may serve as an effective complementary pretraining objective. Second, visual pretraining opens a potential path toward a more scalable pretraining paradigm based on highly compressed visual corpora. This suggests that foundation models could be trained primarily on large-scale visual streams, with only lightweight text alignment, to match the performance of text-pretrained models. Future work should investigate whether this paradigm enables efficient training at larger scales while preserving strong textual and multimodal capabilities.

4. Methods

Overview. VP extends text continued pretraining by introducing next visual latent prediction on raw scientific document pages. Each page is encoded by a frozen vision tower. Retained foreground features are then projected into the LLM's hidden space via a visual projection module. We remove blank background regions, keep foreground patches, and order the remaining features in raster-scan order. Positional encodings are preserved to maintain spatial layout. The resulting sparse visual sequence is fed into the same autoregressive backbone that processes text tokens. The model is trained to predict the next foreground patch feature under causal image-only masking.

Sparse document representation Given a rendered document page $\mathcal{I}$, the frozen vision tower E_{v} produces a sequence of visual features

$$\mathcal{Z} = E_{\mathrm{v}}(\mathcal{I}) = (z_1, \ldots, z_N). \tag{1}$$

Document pages contain large blank regions, such as margins and whitespace. We therefore compute a foreground mask using simple patch-level statistics, including pixel variance and average luminance, and retain only non-blank patches. The retained features are ordered in raster scan to form a sparse foreground sequence

$$\mathcal{U} = \mathrm{Raster}\{z_i : m_i = 1\} = (u_1, \ldots, u_L), L \ll N. \tag{2}$$

This sparse representation keeps foreground document content in page order while substantially shortening the visual context. Each foreground feature is projected into the LLM hidden space through a learned linear projection. Position indices are re-assigned according to the raster order. The exact foreground filtering rule and sequence-packing mask are described in Supplementary Section A.

Next visual latent prediction For each document image, we first extract a sequence of frozen visual latents and retain the foreground tokens. These visual latents are mapped into the LLM embedding space by an input projection module and then fed to the LLM under a causal attention mask over visual positions. Given the LLM hidden state at position t, an output projection head maps it back to the frozen visual-latent space to produce a prediction $\hat{\mathbf{z}}_{t+1}$. The prediction is trained to match the next foreground visual latent $\mathbf{z}_{t+1}$ using a contrastive loss with in-batch negatives. This forms an autoregressive visual-latent prediction objective. It follows the next-token prediction structure of language modeling, but the targets are continuous visual latents rather than discrete text tokens.

We train the visual stream using a next-visual-latent prediction objective. For a batch of predicted–target pairs, let p_{ij} be the softmax probability of matching the prediction at position i to target feature j, computed from cosine similarities with temperature τ. The visual pretraining loss is

$$\mathcal{L}_{\mathrm{VP}} = -\frac{1}{|\mathcal{B}|} \sum_{i \in \mathcal{B}} \log p_{ii}. \tag{3}$$

Other visual features in the batch serve as negatives. This objective encourages the model to predict the correct next document feature while distinguishing it from other patches. The expanded InfoNCE [26] formulation and the definition of the in-batch matching probabilities are provided in Supplementary Section A.

Joint text and visual pretraining The final training objective combines standard text next-token prediction with next visual latent prediction:

$$\mathcal{L} = \lambda_{\mathrm{text}} \mathcal{L}_{\mathrm{CE}} + \lambda_{\mathrm{vis}} \mathcal{L}_{\mathrm{VP}}, \tag{4}$$

where $\mathcal{L}_{\mathrm{CE}}$ is the autoregressive cross-entropy loss on text tokens and $\mathcal{L}_{\mathrm{VP}}$ is defined in equation (3). Text and visual examples are interleaved during training according to a fixed mixing ratio. We update the LLM, the visual input projection and the prediction head, while keeping the visual encoder frozen. Multiple foreground sequences are packed into fixed-length contexts, with sequence boundaries tracked to prevent cross-sample attention. Additional details on the training corpus, architecture, optimization setup and evaluation protocol are provided in Supplementary Section D.

Training setup We implement continued pretraining using the XTuner framework [6]. During visual pretraining, we optimize the LLM and the lightweight prediction projector that maps LLM hidden states to the frozen vision-representation space, while keeping the vision tower frozen. For both TP and VP, we keep the non-PDF text corpus, starting checkpoint, optimization recipe, and SFT stage fixed. The only controlled difference is the representation of the additional scientific-PDF corpus. In TP, the PDF pages are converted into MinerU2.5-parsed text, yielding approximately 80B text tokens. In VP, the same PDF pages are rendered as images and filtered into sparse foreground visual sequences, yielding approximately 20B visual tokens at the main resolution. Thus, the total CPT token budgets differ (180B for TP and 120B for VP) because the same matched PDF corpus is represented more compactly in the visual stream, not because VP uses a different document source. All comparisons therefore match the underlying PDF documents while allowing the token count to reflect the chosen representation. The resulting checkpoints are then trained with the same SFT recipe and evaluated using the same benchmark protocols.

Evaluation protocol As shown in Tables 1 and 2, we evaluate VP and the baselines in a zero-shot setting after SFT initialized from CPT. The evaluations in Table 1 use CoT prompting with the instruction "think step by step", whereas those in Table 2 use a direct-answer template without explicit reasoning guidance. Unless otherwise specified, Table 1 reports average pass@8 for GPQA, the average score over 32 runs for AIME-25, and pass@1 for MMLU-Pro and HLE. All benchmarks in Table 2 are reported with pass@1. By default, we extract answers via rule-based matching and grade them automatically. When extraction fails, we use GPT-4o [11] for judgment.

References

[1] Lawrence W. Barsalou. Perceptual symbol systems. *Behavioral and Brain Sciences*, 22(4):577–660, 1999. doi: 10.1017/S0140525X99002149. 1

[2] Lawrence W. Barsalou. Grounded cognition. *Annual Review of Psychology*, 59:617–645, 2008. doi: 10.1146/annurev.psych.59.103006.093639. 1

[3] Lukas Blecher, Guillem Cucurull Preixens, Thomas Scialom, and Robert Stojnic. Nougat: Neural optical understanding for academic documents. In *International Conference on Learning Representations*, volume 2024, pp. 37646–37663, 2024. 1

[4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 1

[5] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 24185–24198, 2024. 1

[6] XTuner Contributors. Xtuner: A toolkit for efficiently fine-tuning llm. https://github.com/InternLM/xtuner, 2023. 4

[7] Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020. 1

[8] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. 1, 2

[9] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, DDL Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 10, 2022. 1

[10] Minyoung Huh, Brian Cheung, Tongzhou Wang, and Phillip Isola. The platonic representation hypothesis. *arXiv preprint arXiv:2405.07987*, 2024. 1, 2

[11] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024. 4

[12] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020. 1

[13] Geewook Kim, Teakgyu Hong, Moonbin Yim, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. Donut: Document understanding transformer without ocr. *arXiv preprint arXiv:2111.15664*, 7(15):2, 2021. 1, 3

[14] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International conference on machine learning*, pp. 3519–3529. PMlR, 2019. 2, D

[15] Jill H. Larkin and Herbert A. Simon. Why a diagram is (sometimes) worth ten thousand words. *Cognitive Science*, 11(1):65–100, 1987. doi: 10.1016/S0364-0213(87)80026-5. 1

[16] Kenton Lee, Mandar Joshi, Iulia Raluca Turc, Hexiang Hu, Fangyu Liu, Julian Martin Eisenschlos, Urvashi Khandelwal, Peter Shaw, Ming-Wei Chang, and Kristina Toutanova. Pix2struct: Screenshot parsing as pretraining for visual language understanding. In *International Conference on Machine Learning*, pp. 18893–18912. PMLR, 2023. 1, 3

[17] Melody Li, Kumar Krishna Agrawal, Arna Ghosh, Komal Teru, Adam Santoro, Guillaume Lajoie, and Blake Richards. Tracing the representation geometry of language models from pretraining to post-training. *Advances in Neural Information Processing Systems*, 38:54691–54724, 2026. 2

[18] Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, and Kaiming He. Autoregressive image generation without vector quantization. *Advances in Neural Information Processing Systems*, 37:56424–56445, 2024. C

[19] Victor Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Y Zou. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. *Advances in Neural Information Processing Systems*, 35:17612–17625, 2022. 2

[20] Zhiqiu Lin, Xinyue Chen, Deepak Pathak, Pengchuan Zhang, and Deva Ramanan. Revisiting the role of language priors in vision-language models. *arXiv preprint arXiv:2306.01879*, 2023. B

[21] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023. 1

[22] Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations*, volume 2024, pp. 23439–23554, 2024. 2

[23] Ahmed Masry, Mohammed Saidul Islam, Mahir Ahmed, Aayush Bajaj, Firoz Kabir, Aaryaman Kartha, Md Tahmid Rahman Laskar, Mizanur Rahman, Shadikur Rahman, Mehrad Shahmohammadi, et al. Chartqapro: A more diverse and challenging benchmark for chart question answering. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 19123–19151, 2025. 2, 2

[24] Mathematical Association of America. American invitational mathematics examination (AIME). 2025. URL https://maa.org/math-competitions/american-invitational-mathematics-examination-aime. Administered by the MAA as part of the AMC competition series. 2

[25] Meta AI. Llama 3.2 vision model card. https://huggingface.co/meta-llama/Llama-3.2-11B-Vision, 2024. Accessed: 2026-06-02. 1, 2, 2b

[26] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 4

[27] Keiran Paster, Marco Dos Santos, Zhangir Azerbayev, and Jimmy Ba. Openwebmath: An open dataset of high-quality mathematical web text. In *International Conference on Learning Representations*, volume 2024, pp. 20357–20379, 2024. 1

[28] Long Phan, Alice Gatti, Ziwen Han, Nathaniel Li, Josephina Hu, Hugh Zhang, Chen Bo Calvin Zhang, Mohamed Shaaban, John Ling, Sean Shi, et al. Humanity's last exam. *arXiv preprint arXiv:2501.14249*, 2025. 2

[29] Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5. 1, 2, 2b

[30] David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. Gpqa: A graduate-level google-proof q&a benchmark. *arXiv preprint arXiv:2311.12022*, 2023. 2

[31] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. URL http://jmlr.org/papers/v9/vandermaaten08a.html. 1

[32] Bin Wang, Chao Xu, Xiaomeng Zhao, Linke Ouyang, Fan Wu, Zhiyuan Zhao, Rui Xu, Kaiwen Liu, Yuan Qu, Fukai Shang, et al. Mineru: An open-source solution for precise document content extraction. *arXiv preprint arXiv:2409.18839*, 2024. 1, D

[33] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 1

[34] Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *arXiv preprint arXiv:2406.01574*, 2024. 2

[35] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*, 2022. 2

[36] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. 1, 2

[37] Xiang Yue, Tianyu Zheng, Yuansheng Ni, Yubo Wang, Kai Zhang, Shengbang Tong, Yuxuan Sun, Botao Yu, Ge Zhang, Huan Sun, et al. Mmmu-pro: A more robust multi-discipline multimodal understanding benchmark. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15134–15186, 2025. 2

[38] Jiajie Zhang. The nature of external representations in problem solving. *Cognitive Science*, 21(2):179–217, 1997. doi: 10.1207/s15516709cog2102_3. 1

[39] Yuhao Zhou, Yiheng Wang, Xuming He, Ruoyao Xiao, Zhiwei Li, Qiantai Feng, Zijie Guo, Yuejin Yang, Hao Wu, Wenxuan Huang, et al. Scientists' first exam: Probing cognitive abilities of mllm via perception, understanding, and reasoning. *Advances in Neural Information Processing Systems*, 38, 2026. 2

Supplementary overview. We provide additional analyses and implementation details that complement the main text.

- **Section A (Detailed Visual-Pretraining Pipeline).** We provide the full mathematical formulation of the visual pretraining pipeline, including frozen visual feature extraction, foreground-token filtering, causal visual prediction, contrastive next-visual-latent loss, and sequence packing.
- **Section B (PPL-Based Image-to-Text Retrieval).** We provide an alternative generative evaluation of cross-modal alignment through perplexity-based image-to-text retrieval, demonstrating that visual pretraining substantially enhances retrieval accuracy.
- **Section C (Further Studies: Visual Pretraining with Generative Decoder).** We explore augmenting visual pretraining with an explicit generative decoder that provides pixel-level reconstruction supervision. We find that this variant incurs substantially higher training cost while achieving performance comparable to the decoder-free visual pretraining formulation.
- **Section D (Implementation and Evaluation Details).** We describe the training corpus, model architecture, optimization setup, evaluation protocol, and cross-modal alignment analysis used in the main experiments.

A. Detailed Visual-Pretraining Pipeline

This section provides the mathematical details of the visual-pretraining pipeline used in VP. The main text describes the method at a high level; here we specify the construction of sparse document tokens, the causal visual-prediction objective, and the sequence-packing procedure used in training.

Frozen visual feature extraction. Given a rendered document page $\mathcal{I} \in \mathbb{R}^{H\times W\times 3}$, a frozen ViT encoder ϕ_{ViT} with patch size p first maps the page into a grid of patch features:

$$Y = \phi_{\mathrm{ViT}}(\mathcal{I}) = (y_1, \ldots, y_{N_0}), \qquad y_i \in \mathbb{R}^{c_v}. \tag{5}$$

A frozen spatial merger M then groups neighbouring patch features and maps them into the visual feature space used as the prediction target:

$$\mathcal{Z} = M(Y) = (z_1, \ldots, z_N), \qquad z_i \in \mathbb{R}^{d_v}. \tag{6}$$

The ViT encoder and the spatial merger are fixed throughout training. Thus, the visual stream learns to predict stable document features rather than reconstructing pixels.

Foreground-token filtering. Document pages contain large blank regions, such as margins and whitespace. To avoid spending context length on background patches, we compute a patch-level foreground mask before constructing the visual sequence. For each raw image patch, we compute its pixel variance σ_i^2 and average luminance ℓ_i. A patch is treated as background when it has low variance and near-blank luminance:

$$b_i = \mathbf{1}\left[\sigma_i^2 < \tau_\sigma \ \wedge\ \left(\ell_i > \tau_\ell^+ \vee \ell_i < \tau_\ell^-\right)\right], \tag{7}$$

where $b_i = 1$ indicates background. After spatial merging, the mask is max-pooled over each merged block, so a merged visual token is retained if any of its constituent patches contains foreground content:

$$m_j = \max_{i\in\mathcal{P}(j)} (1 - b_i), \tag{8}$$

where $\mathcal{P}(j)$ denotes the set of raw patches belonging to merged token j.

The retained visual features are ordered in raster scan:

$$\mathcal{U} = \mathrm{Raster}\{z_j : m_j = 1\} = (u_1, \ldots, u_L), \qquad L \ll N. \tag{9}$$

This produces a sparse document sequence that preserves page order while removing most blank regions.

Projection into the LLM space. Each foreground visual feature is projected into the LLM hidden space by a learned linear map:

$$x_i = W_{\mathrm{in}} u_i, \qquad x_i \in \mathbb{R}^d. \tag{10}$$

Position indices are assigned according to the raster order after foreground filtering. This allows the LLM to process the page as an ordered visual sequence while avoiding position allocation to removed background regions.

Causal visual prediction. The projected sequence $(x_1, \ldots, x_L)$ is fed into the shared autoregressive LLM backbone with a causal attention mask over image positions. The hidden state at position i is used to predict the next foreground feature:

$$h_i = \Phi_{\mathrm{LLM}}(x_{\le i}), \qquad \hat{u}_{i+1} = \psi(h_i), \tag{11}$$

where Φ_{LLM} is the LLM backbone and ψ is a lightweight MLP prediction head. In our implementation,

$$\psi(h) = W_2\, \mathrm{GELU}(W_1 h). \tag{12}$$

The target for $\hat{u}_{i+1}$ is the frozen visual feature u_{i+1}. Therefore, the task is a continuous analogue of language-model next-token prediction.

Contrastive next-visual-latent loss. For a batch of valid predicted–target pairs $\mathcal{B}$, we compute cosine-similarity logits:

$$s_{ij} = \frac{\cos(\hat{u}_{i+1}, u_{j+1})}{\tau}, \tag{13}$$

where τ is a temperature parameter. The matching probability is obtained by a softmax over target features in the batch:

$$p_{ij} = \mathrm{softmax}_{j \in \mathcal{B}}(s_{ij}). \tag{14}$$

The visual pretraining loss is then

$$\mathcal{L}_{\mathrm{VP}} = -\frac{1}{|\mathcal{B}|} \sum_{i \in \mathcal{B}} \log p_{ii}. \tag{15}$$

This objective encourages each predicted feature to match the correct next foreground token while using other visual features in the batch as negatives.

Joint optimization. Visual pretraining is combined with standard text next-token prediction:

$$\mathcal{L} = \lambda_{\mathrm{text}} \mathcal{L}_{\mathrm{CE}} + \lambda_{\mathrm{vis}} \mathcal{L}_{\mathrm{VP}}. \tag{16}$$

Only the LLM parameters, the visual input projection W_{in}, and the prediction head ψ are updated. The ViT encoder and spatial merger remain frozen, providing a fixed visual target space.

Sequence packing and attention masking. To improve training efficiency, multiple sparse foreground sequences are packed into a fixed-length context. Let $q(a)$ denote the document sequence to which packed position a belongs. We use a block-causal attention mask:

$$A_{ab} = \begin{cases} 0, & q(a) = q(b) \text{ and } b \le a, \\ -\infty, & \text{otherwise.} \end{cases} \tag{17}$$

This prevents visual tokens from attending to future tokens or to tokens from other packed document pages. Prediction targets are also restricted within the same original foreground sequence, so the model never predicts across document boundaries.

B. PPL-Based Image-to-Text Retrieval

We provide an alternative, generative evaluation of cross-modal alignment by casting it as image-to-text retrieval scored by conditional perplexity.

Evaluation setting. We sample 100 random document image–text pairs and, for each (image v_i, text t_j) combination, compute a matching score defined following Lin et al. [20] as the pointwise mutual information estimated via conditional perplexity, $\text{Score}(i, j) = \mathcal{L}(t_j \mid v_i) - \mathcal{L}(t_j)$, where $\mathcal{L}(t_j \mid v_i)$ is the cross-entropy loss of generating text t_j conditioned on visual representation v_i and $\mathcal{L}(t_j)$ is the unconditional text prior; a lower score indicates stronger alignment. We report Recall@K and Mean Reciprocal Rank (MRR).

Findings. As shown in Table 3, visual pretraining yields a substantial improvement: R@1 increases from 64.0% to 99.0% and MRR rises from 78.2 to 99.5. Analysis of the 100×100 score matrix reveals that in the original model, off-diagonal scores exhibit high variance ($\sigma = 0.168$), causing certain texts to act as "hub" attractors that receive spuriously low scores regardless of the input image. After visual pretraining, the off-diagonal variance drops to $\sigma = 0.043$, effectively eliminating the hub phenomenon and enabling near-perfect discrimination.

Table 3: Generative image-to-text retrieval with conditional perplexity. Each image is matched against 100 candidate texts using the PMI-style score defined in Section B; higher Recall@K and MRR indicate better cross-modal compatibility.

Metric	Text Pretraining	VP (Ours)	Δ
R@1	64.0	99.0	+35.0
R@5	92.0	100.0	+8.0
MRR	78.2	99.5	+21.3

C. Visual Pretraining with Generative Decoder

Motivation. A natural extension of visual pretraining is to equip the LLM with an explicit *generative decoder* that decodes visual latents back to pixels, thereby providing pixel-level reconstruction supervision. In our exploratory experiments, we instantiate this design by attaching a MAR-style autoregressive decoder [18] followed by a frozen VAE decoder to the LLM's hidden states, yielding a full encode–decode pipeline as illustrated in Figure 5. The resulting pixel reconstruction loss $\mathcal{L}_{\text{MSE}}$ offers fine-grained, pixel-level supervision that is absent in the pure latent-prediction formulation used in the main text.

Comparison with generative visual pretraining. Despite the extra supervision signal, the generative-decoder design incurs a substantial increase in training cost. Specifically, (1) the MAR decoder introduces additional trainable parameters ($\sim$300M in our setup) and computation; (2) the diffusion-style latent loss $\mathcal{L}_{\text{diff}}$ and the pixel MSE loss $\mathcal{L}_{\text{MSE}}$ must both be computed and back-propagated through the decoder stack at every visual step; and (3) the overall training throughput drops by roughly 30%–40% compared with the decoder-free variant. In our experiments, the generative-decoder variant requires approximately $1.4\times$ the training time to reach the same convergence criterion as the decoder-free formulation. These overheads become significant when scaling to large document corpora.

We further compare the generative-decoder variant (denoted **VP with Decoder**) against the decoder-free visual pretraining (denoted **VP w/o Decoder**) on the same matched document sources. As summarized in Table 4, both visual pretraining variants outperform the Base model and the text-only pretraining baseline across all evaluated reasoning benchmarks. Crucially, the pixel-level loss does *not* translate into consistently stronger language-reasoning performance: the generative-decoder variant achieves comparable or only marginally better scores than the decoder-free formulation, suggesting that the extra pixel supervision yields diminishing returns for downstream reasoning.

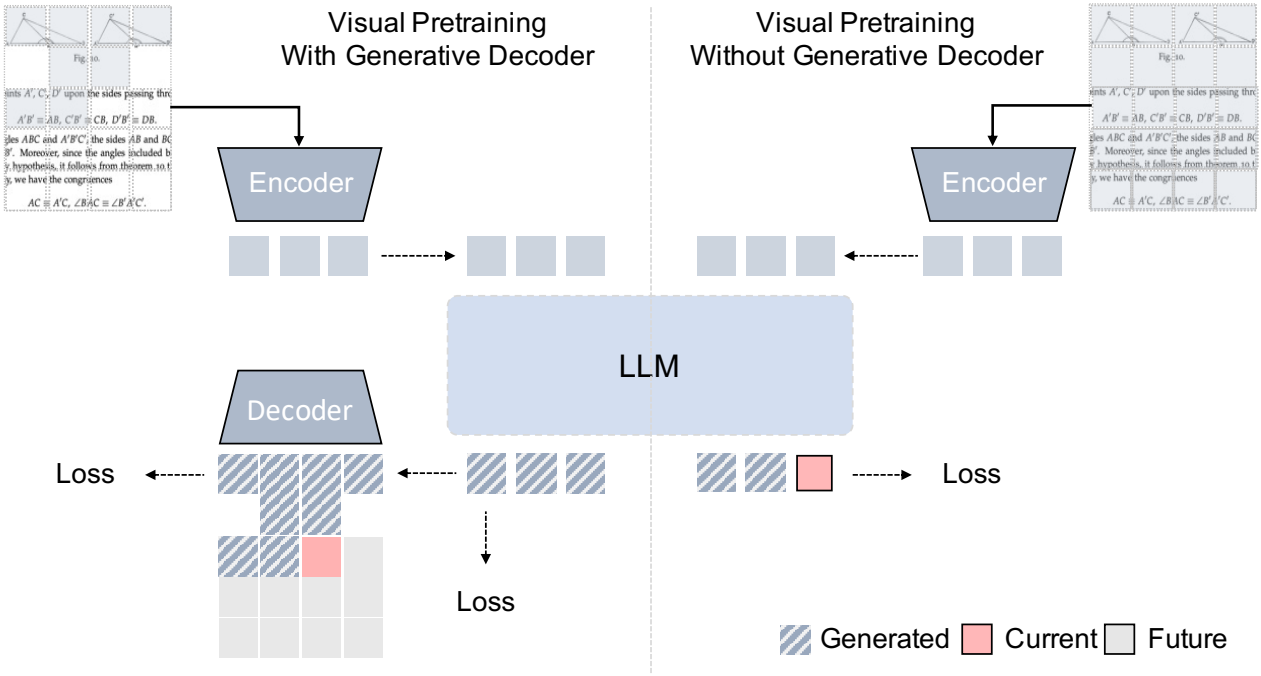

Figure 5: **Architecture comparison between the generative-decoder variant and decoder-free visual pretraining.** (Left) The generative-decoder variant (VP with Decoder) adds a MAR decoder and a frozen VAE decoder to reconstruct pages from LLM-refined latents, providing additional pixel-level supervision at the cost of extra parameters and computation. (Right) Our decoder-free formulation (VP w/o Decoder) predicts foreground patch latents autoregressively without pixel reconstruction.

Table 4: **Decoder-free visual pretraining versus the generative-decoder variant.** All methods use the same matched document sources and are evaluated after the same SFT protocol. The decoder-free formulation attains comparable or better reasoning performance without the additional reconstruction decoder.

Method	MMLU-Pro	GPQA	AIME
Base	79.86	69.57	74.90
Text pretraining	81.32	74.94	74.48
VP with Decoder	81.87	75.44	76.46
VP w/o Decoder	81.69	75.95	76.98

Conclusion. Taken together, these exploratory results suggest that pixel-level generation supervision offers limited marginal benefit for language reasoning, at a non-negligible computational cost. Consequently, the main text adopts the *decoder-free* visual pretraining formulation, which attains comparable downstream performance through a substantially simpler and more efficient pipeline.

D. Additional Implementation and Evaluation Details

This section provides implementation and evaluation details that complement the main Methods. We first describe the training data, model architecture, optimization setup, and benchmark protocol. We then give the full construction of the cross-modal alignment analysis reported in Table 2 and discussed in Section 2.

Training corpus. The training data consist of three components: a text corpus for standard continued pretraining, a scientific-PDF corpus for matched visual and text pretraining, and SFT data for instruction tuning. For the matched text-pretraining baseline, each PDF page is converted into text with MinerU2.5 [32]. For VP, the same PDF pages are rendered as images and consumed as unlabeled visual supervision. For the

pretraining of Qwen, the actual preprocessing pipeline yields approximately 20B retained visual tokens for VP and approximately 80B parsed text tokens for TP from the same PDF pages. This matched construction controls the document source and changes only the representation of the scientific-PDF corpus.

The VP image records are stored as JSONL entries containing page-image paths and optional precomputed latent-feature paths. During loading, each page is converted to RGB, resized to a square at a fixed maximum resolution, and normalized before being passed through the model processor. Unless otherwise stated, pages are sampled without filtering by crop coordinates. Blank regions are removed by a foreground mask computed from patch variance and luminance, using variance threshold 0.02, high-luminance threshold 0.95, and low-luminance threshold 0.15. A merged visual token is retained if any of its constituent raw patches is foreground.

Model architecture. VP uses the same LLM backbone as the corresponding text-pretraining baseline. In the Qwen3.5 implementation, the visual pathway is initialized from the corresponding Qwen3.5/Qwen-VL checkpoint. The vision tower is a ViT-style encoder with 27 layers, hidden size 1152, 16 attention heads, patch size 16, and spatial merge size 2. The vision tower is frozen during VP. The foreground visual features are mapped into the LLM hidden space through a visual projector; in the main VP runs, this projector and the LLM are trainable, while the frozen vision features serve as the prediction targets. A two-layer MLP prediction head maps LLM hidden states back to the frozen visual-feature space for next visual latent prediction.

The main formulation of VP is decoder-free: it predicts frozen visual features and does not reconstruct pixels. This avoids the additional computational cost and optimization complexity of a pixel-level image decoder. We separately study a generative-decoder variant in Section C.

Training hyperparameters. Text and visual examples are interleaved during continued pretraining according to a fixed mixing ratio. In our implementation, one VP batch is attached to every text-training step. The text branch is optimized with standard autoregressive cross-entropy, while the visual branch is optimized with the next visual latent contrastive loss defined in Equation (3). We use InfoNCE with temperature $\tau = 0.07$ and set the VP loss weight to 0.1 in the main Qwen3.5 runs. The LLM backbone, visual projector, and prediction head are updated jointly, while the vision tower remains frozen throughout training.

Multiple sparse foreground sequences are packed into fixed-length contexts, and sequence boundaries are tracked to prevent attention across different document pages. The detailed mathematical formulation of the visual pipeline, including foreground filtering, projection, causal masking, and sequence packing, is provided in Section A.

For the Qwen3.5 runs, text sequences are hard-packed to 32,768 tokens and trained for 12,000 steps with global batch size 1024, AdamW learning rate 3×10^{-5}, weight decay 0.1, a 1,000-step warmup, and linear decay to 10^{-6}. The corresponding VP head uses AdamW with learning rate 4×10^{-5} and the same optimizer betas and weight decay as the text optimizer. The visual VP context is capped at 8,192 retained foreground tokens for the main setting.

Training cost. Across the main Qwen and Llama VP/TP continued-pretraining runs, each run typically required approximately 1.5–3 days of wall-clock time on a distributed cluster with 128 accelerators, depending on the backbone, token budget, context length, and data modality. For example, the main Qwen3.5-35B-A3B continued-pretraining run required approximately 36 hours, followed by approximately 10 hours for the SFT stage. The exact accelerator model is not disclosed due to institutional constraints.

Evaluation protocol. All models are evaluated in a zero-shot setting after SFT initialized from CPT. For text-only reasoning benchmarks in Table 1, we use CoT prompting with the instruction “think step by step”. Unless otherwise specified, we report average pass@8 for GPQA, the average score over 32 runs for AIME-25, and pass@1 for MMLU-Pro and HLE.

For multimodal benchmarks in Table 2, we use a direct-answer template without explicit reasoning guidance and report pass@1. These evaluations test whether the same unlabeled visual-document pretraining signal that improves language reasoning also transfers to multimodal reasoning.

Cross-modal alignment analysis. This paragraph details the cross-modal alignment evaluation reported in Table 2 and qualitatively visualized in Figure 1.

Setting. We construct a set of $N{=}100$ matched document image–text pairs. Each pair is obtained by rendering a scientific PDF page as a RGB image and using the OCR-parsed text from the same page as its textual counterpart. The pairs are drawn from a held-out set of scientific PDFs collected from publicly accessible repositories.

For each pair, the image is processed by the visual pathway and the text by the tokenizer of the shared backbone. We collect hidden states from the last Transformer layer and mean-pool over foreground visual tokens and non-padding text tokens, obtaining one embedding per modality. This yields paired embeddings

$$\{(v_i, t_i)\}_{i=1}^{N},$$

where v_i and t_i denote the visual and textual representations of the same document page. The same procedure is applied before and after VP, enabling a direct comparison of representation alignment.

Metric definitions. We measure alignment from three complementary perspectives: global distance, structural geometry, and local neighbourhood consistency.

Global alignment is measured by centroid separation and pairwise cosine similarity. Centroid separation is defined as

$$\|\bar{v} - \bar{t}\|_2,$$

where $\bar{v} = \frac{1}{N}\sum_i v_i$ and $\bar{t} = \frac{1}{N}\sum_i t_i$ are the modality-wise mean embeddings. Pairwise cosine similarity is computed as

$$\frac{1}{N}\sum_{i=1}^{N} \cos(v_i, t_i).$$

Structural alignment is measured by linear Centered Kernel Alignment (CKA) [14]. Let $V, T \in \mathbb{R}^{N\times d}$ denote the centered embedding matrices of the visual and textual representations. We compute

$$\mathrm{CKA}(V, T) = \frac{\|V^\top T\|_F^2}{\|V^\top V\|_F \, \|T^\top T\|_F}.$$

Linear CKA compares the relational geometry of the two embedding spaces and is invariant to orthogonal transformations and isotropic scaling.

Local alignment is measured by mutual k-nearest-neighbour overlap:

$$A_k = \frac{1}{Nk}\sum_{i=1}^{N} \left|\mathcal{N}_k^V(i) \cap \mathcal{N}_k^T(i)\right|,$$

where $\mathcal{N}_k^V(i)$ and $\mathcal{N}_k^T(i)$ are the k nearest neighbours of sample i in the visual and textual embedding spaces under cosine distance. We report $k \in \{1, 5, 10\}$.

Findings. The three groups of metrics provide complementary evidence that VP improves cross-modal alignment without image–text pair supervision. Globally, centroid separation drops by 60% from 1.665 to 0.661, while pairwise cosine similarity rises from 0.631 to 0.907. This indicates that paired visual and textual embeddings are brought into a more compatible region of representation space.

Structurally, linear CKA increases from 0.657 to 0.745, showing that the improvement is not merely a shift in average distance but also a change in the relational geometry of the two modalities. Locally, mutual k-NN overlap improves at all tested values of k, with the largest gain at the most stringent setting: $k{=}1$ increases from 0.140 to 0.310, while $k{=}5$ increases from 0.288 to 0.420 and $k{=}10$ from 0.395 to 0.496.

This pattern argues against trivial representation collapse. Collapse would tend to inflate neighbourhood overlap uniformly, whereas VP produces the strongest improvement at the finest local scale. Together, the global, structural, and local metrics indicate that visual pretraining reshapes the shared representation space in a way that makes visual and textual document states more compatible.