Vision-Text Augmentation: Visual Composition as an Optimization Axis for Language Tasks

Anonymous ACL submission

Abstract

Recent work on Vision-Text Compression (VTC) focuses on reducing token counts for long-context efficiency, but the vast design space of visual rendering remains largely unexploited. We study a complementary direction: **Vision-Text Augmentation (VTA)**, where text is rendered into image space and varied across visual composition factors such as font, spacing, and layout to improve VLM performance. We first conduct a comparative study of two meaning-preserving variation spaces: text-level variations (e.g., paraphrasing) for LLMs, and visual composition variations for VLMs. Our evaluation on information retrieval (IR) and question answering (QA) reveals a clear asymmetry: text-level variations fail to reliably improve LLMs, whereas visual composition variations yield substantial gains for VLMs, at times even outperforming LLMs. Motivated by this finding, we introduce **VTAOpt**, a model-agnostic Bayesian optimization framework that discovers effective rendering configurations without model tuning. With VTAOpt, VLMs outperform the best text-based models on most benchmarks, with gains of up to 13.7 nDCG points on IR and 29.4 points on QA. Our findings show that the visual rendering space affords VLMs greater performance potential than the text space affords LLMs, opening VTA as a new optimization axis for NLP beyond prompt engineering [1].

1 Introduction

Long-context understanding is a fundamental challenge in Transformer based Large Language Models (LLMs) due to the quadratic complexity of the self-attention mechanism (Vaswani et al., 2017). To address this, numerous methods have been proposed to improve efficiency (Press et al., 2021; Su et al., 2024; Child et al., 2019; Shazeer, 2019; Ainslie et al., 2023; Dao et al., 2022). Recent findings in OCR and Vision Language Modeling (VLMs) research suggest a promising new direction called Visual Text Compression (VTC) (Wei et al., 2025; Cheng et al., 2025; Zhao et al., 2025), where even for text-only tasks, documents are rendered as images and processed by vision encoders. While prior VTC work treats text-to-image rendering as an efficiency mechanism that trades off accuracy, we view this pipeline as an unexploited optimization space for model performance. We study a complementary direction, **Vision-Text Augmentation (VTA)**: by rendering text into image space, VLMs gain an extra degree of freedom beyond linguistic variation (e.g., paraphrasing), as visual composition (e.g., text size, spacing, and layout)

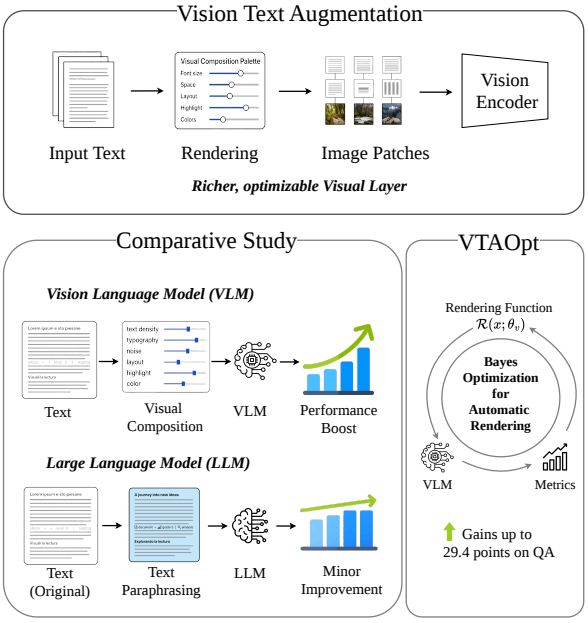


Figure 1: Overview of our approach. **Top:** VTA renders text into image space with varied visual composition. **Bottom left:** Visual composition boosts VLMs; paraphrasing does not reliably improve LMs. **Bottom right:** VTAOpt automatically discovers effective rendering configurations via Bayesian optimization.

[1] Dataset and Project code will be made public upon acceptance.

can be freely varied to affect the resulting visual tokens. VTA raises two research questions: **RQ1:** *Compared to LLMs, can VLMs' richer input space be exploited to improve downstream performance?* **RQ2:** *If so, can the optimal visual composition be discovered automatically?* Figure 1 illustrates our approach.

To answer **RQ1**, we design a comparative study across information retrieval (IR) and question answering (QA) tasks (Section 4). For VLMs, we render the same text under different visual compositions and measure the effect on performance; for LLMs, we paraphrase the input while preserving its meaning and apply the same evaluation. Our results reveal a clear asymmetry: simply varying visual composition yields gains of up to 15 nDCG points on IR and 9 accuracy points on QA over default rendering, whereas paraphrasing does not produce significant improvements for LLMs. This suggests that the rendering pipeline is not merely a preprocessing step but an optimizable representation layer.

To answer **RQ2**, we introduce **VTAOpt**, a model-agnostic Bayesian optimization framework that automatically discovers effective visual composition configurations for a given VLM without modifying model weights (Section 5). The rendering parameter space is high-dimensional, combinatorial, and model-dependent, making manual tuning infeasible. With VTAOpt, VLMs outperform the strongest text-based models on most benchmarks, with gains of up to 13.7 nDCG points on IR and 29.4 points on QA.

In summary, our contributions are twofold: (1) We conduct a systematic comparison of input representation effects across VLMs and LLMs, showing that visual composition significantly improves VLM performance while paraphrasing does not reliably benefit LLMs. (2) We introduce **VTAOpt**, a model-agnostic Bayesian optimization framework that discovers optimal rendering configurations without model tuning, demonstrating that optimized VLMs can outperform LLMs on both IR and QA tasks.

2 Related Work

Vision Language Models Recent VLMs largely follow two directions: contrastive alignment and LLM-centric architectures. Contrastive models such as CLIP (Radford et al., 2021) and SigLIP (Zhai et al., 2023) learn shared image–text embedding spaces with similarity-based objectives, enabling strong zero-shot transfer. A parallel line aligns pretrained visual encoders with LLMs via lightweight projection modules; Frozen (Tsimpoukelli et al., 2021) and BLIP-2 (Li et al., 2023) are early examples. Newer systems, including Qwen3-VL (Bai et al., 2025) and InternVL3.5 (Wang et al., 2025), scale this design to improve multimodal reasoning on diverse benchmarks.

Task	Dataset	#Q	#Ctx	#Wds (Min/Mean/Max)
IR	MultiReQA (Guo et al., 2021)	161	2220	500/687/799
IR	NarrativeQA (Kočiský et al., 2018)	19965	700	500/686/999
IR	Google NQ (Kwiatkowski et al., 2019)	300	296	506/3619/25178
QA	Memory (Zhao et al., 2025)	200	200	9804/16787/19405
QA	Retrieval (Zhao et al., 2025)	200	200	833/9606/26598
QA	Reasoning (Zhao et al., 2025)	200	200	728/6725/27689

Table 1: Datasets for information retrieval (IR) and question answering (QA). Context length reported as Min/Mean/Max words.

Vision-Text Compression Vision-Text Compression (VTC) renders long text into compact visual inputs to reduce the compute and memory costs of long-context modeling (Xing et al., 2025; Wang et al., 2024; Li et al., 2025). VIST uses a slow–fast scheme to skim low-salience context, achieving a 2.3× token reduction with strong in-context accuracy (Xing et al., 2025), while DeepSeek-OCR shows more extreme 2D optical mapping (up to 10×) with 97% decoding precision (Wei et al., 2025). Recent methods such as Glyph (Cheng et al., 2025) and VTC-R1 (Wang et al., 2026) extend VTC to reasoning via reinforcement fine-tuning. VTCBench evaluates retrieval, reasoning, and long-term memory, finding that models handle OCR well but struggle with complex reasoning over compressed visual context (Zhao et al., 2025).

3 Experimental Design

3.1 Datasets

We evaluate on datasets spanning long-document **IR** and **QA**, covering both web-based question answering and long-form narrative understanding. Table 1 summarizes the dataset statistics and splits for both tasks.

3.2 Tasks

IR. We study long-document information retrieval, where a query q is matched against a corpus of documents $\mathcal{C} = \{x_1, \ldots, x_M\}$. Each document is encoded into an embedding, and retrieval is performed by computing cosine similarity scores and

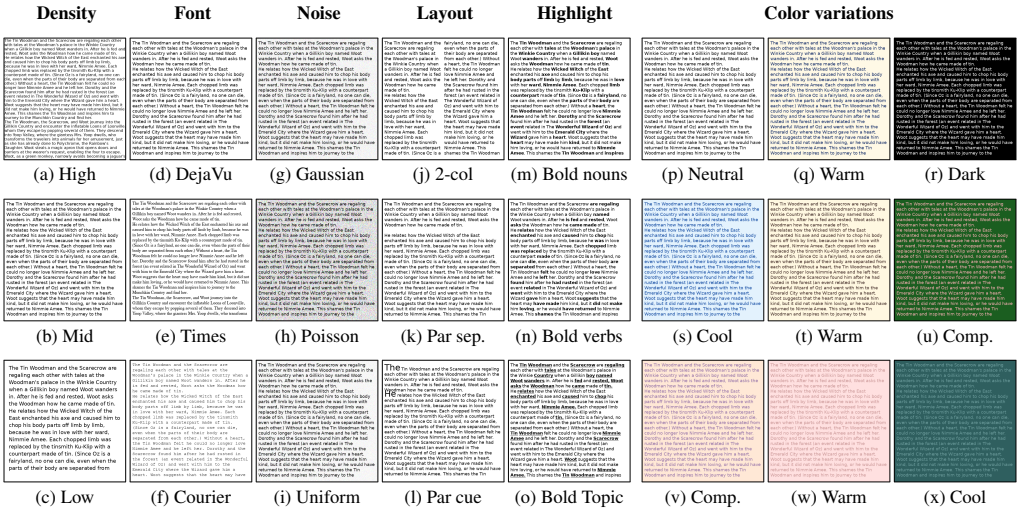


Figure 2: **Visual composition palette.** Left: non-color composition variablestext density (ac), font type (df), noise perturbations (gi), layout (jl), and content-aware highlighting (mo). Right: the 3×3 color and luminance palette (px) organized by contrast level and color family. All examples are rendered from the same underlying text.

ranking documents accordingly.

QA. We follow the VTCBench formulation (Zhao et al., 2025), where a model is given a question q and a long-form context x and generates an answer a. The benchmark comprises three tasks: **Memory** (long-term dialogue recall), **Reasoning** (associative inference over long contexts), and **Retrieval** (needle-in-haystack information extraction).

3.3 Models

We evaluate both text-based and vision-based model families. For VLMs, we use CLIP-Base, CLIP-Large, SigLIP-Base, SigLIP-Large, and ColFlor for IR, and Qwen3-VL and Glyph for QA. For LLMs, we use sentence transformers (MiniLM) and standard text encoders. Full model details are provided in Appendix C.

3.4 Evaluation Metrics

For IR, we report nDCG@20 in main tables, with additional metrics (Recall@1,5,10,20, MRR@20, MAP) provided in Appendix tables. For QA, we follow VTCBench and use ROUGE-L for Memory and Contains-All for Reasoning and Retrieval tasks.

4 Comparing Input Representations

4.1 Problem Setup

We investigate how semantics-preserving transformations of input representations affect model performance across two paradigms: VLMs and LLMs. Given a document x with semantic content $s(x)$, we define a transformation $\mathcal{T}$ as *semantics-preserving* if $s(\mathcal{T}(x)) \approx s(x)$.

Visual Composition for VLMs. We define a rendering function $\mathcal{R}(x; \theta_v)$ that maps text x into an image I under composition parameters θ_v. The VLM encodes this image: $f_{\text{VLM}}(x; \theta_v) = V(\mathcal{R}(x; \theta_v))$. The parameters θ_v control visual properties such as text density, typography, layout, and color.

Text Paraphrasing. We define a paraphrasing function $\mathcal{P}(x; \theta_t)$ that transforms the linguistic surface form according to type θ_t. Because text inputs to both standalone language models and VLMs are processed by their respective language backbones, we refer to all text-evaluating architectures collectively as LLMs. The model encodes the paraphrased text directly: $f_{\text{LM}}(x; \theta_t) = E(\mathcal{P}(x; \theta_t))$, where $E(\cdot)$ denotes the text encoding function of the LLM or VLM language backbone. The transformations include lexical substitution, syntactic reordering, and LLM-based paraphrasing. The detailed variants of these paraphrasing transformations are provided in Section 4.3.1.

Research Question. How sensitive are models to input representation changes, and can alternative configurations improve over defaults? For each model, we report performance under a *Default* configuration (θ^0, i.e. standard rendering for VLMs or original text for LLMs), as well as the *Worst*

and *Best* configurations found across all variations. We then measure two quantities: (1) *degradation* $\Delta = \mathcal{M}(\theta_{\text{worst}}) - \mathcal{M}(\theta^0)$, which captures the lower bound of the variation space; and (2) *improvement* $\Delta^* = \mathcal{M}(\theta_{\text{best}}) - \mathcal{M}(\theta^0)$, which captures the upper bound. We evaluate on IR and QA tasks. Full experimental details are provided in Section 3.

4.2 VLMs Visual Composition

4.2.1 Method

We study document rendering as a configurable visual composition process and examine composition variables spanning text density, typography, noise, layout and spatial structure, content-aware highlighting, and color and luminance design. Figure 2 illustrates this visual composition palette. Detailed definitions and implementation details for each composition variable are provided in Appendix Table 6 and Appendix D.

4.2.2 Results

Table 2 (top) summarizes the impact of visual composition on vision encoder and VLM performance across information retrieval and question answering tasks. We define *Default* as a standard rendering configuration: font size 16, line spacing 2, padding 1, DejaVu font with regular style and single-column layout. *Worst* and *Best* represent the lowest and highest performance observed across all composition settings in our systematic study (full results in Appendix F). Notably, these results are obtained through exhaustive evaluation of predefined composition variants, not through optimization algorithms such as VTAOpt.

The results reveal substantial sensitivity to visual composition. For IR, CLIP-Base exhibits significant upside potential with an improvement (Δ^*) of $+12.6$ nDCG@20 points, while SigLIP-Base shows even greater sensitivity with $\Delta^* = +15.2$ points, alongside minor downside degradations (Δ) of -0.4 and -3.4 points respectively. For question answering, Glyph demonstrates the highest upside sensitivity ($\Delta^* = +8.9$ points), followed by Qwen3-VL at $\Delta^* = +5.3$ points. Importantly, across both tasks, the best-performing visual compositions consistently and significantly outperform the default settings (Δ^* ranging from $+5.3$ to $+15.2$ points) with minimal degradation risks. This indicates that visual composition is not merely a source of random variance but a highly optimizable representation layer that can substantially improve VLM performance.

Model	Def.	Worst	Best	Δ	Δ^*
VISUAL COMPOSITION (Systematic Study)					
IR Tasks (nDCG@20)					
CLIP-Base	66.3	65.9	78.9	-0.4	**+12.6**†
SigLIP-Base	56.3	52.9	71.5	-3.4	**+15.2**†
QA Tasks (VLMs)					
Qwen3-VL	51.1	49.9	56.4	-1.2	**+5.3**†
Glyph	55.9	53.2	64.8	-2.7	**+8.9**†
TEXT PARAPHRASING					
IR Tasks (Text Encoders)					
MiniLM	83.1	80.3	83.7	-2.8	+0.6
bge-L	84.6	81.5	84.6	-3.1	0.0
QA Tasks (LLMs)					
Llama-8b	57.8	10.0	58.5	**-47.8**†	+0.7
GPT-4o-mini	57.2	10.1	57.7	**-47.1**†	+0.5
Qwen3-VL	67.0	12.7	68.8	**-54.3**†	+1.8

Table 2: Impact of visual composition vs. text paraphrasing. Columns arranged from degradation (Δ = Worst$-$Default) to improvement (Δ^* = Best$-$Default). Generative models are evaluated at a temperature of 0 for robust testing. † denotes statistical significance ($p < 0.05$, paired t-test).

4.3 LLMs Paraphrasing

4.3.1 Method

We study how linguistic surface-form transformations affect text encoder performance. Following the visual composition study for VLMs, we design a parallel set of semantics-preserving transformations for text-based models spanning four categories: (1) *Lexical*, with synonym substitution using WordNet; (2) *Syntactic*, with *active* and *passive* voice conversion variants using an LLM and spaCy; (3) *Style*, with LLM-based rewriting variants including *compression*, *expansion*, *formal*, and *entertainment*; and (4) *Highlight*, with variants that insert markers around *nouns*, *verbs*, or *topic* words using IF-IDF routine.

4.3.2 Results

Table 2 (bottom) summarizes the impact of text paraphrasing on LLM performance across IR and QA tasks. For IR, we evaluate text encoders, while for QA, we evaluate both standalone LLMs and the language backbones of VLMs, referring to all text-evaluating architectures collectively as LLMs. The results reveal that paraphrasing cannot significantly improve LLM performance, and improper paraphrasing methods can substantially degrade it, where *Default* denotes the original unmodified text.

For IR tasks, no paraphrasing strategy significantly improves over the original text (*Default*). MiniLM achieves its best performance at 83.7, a

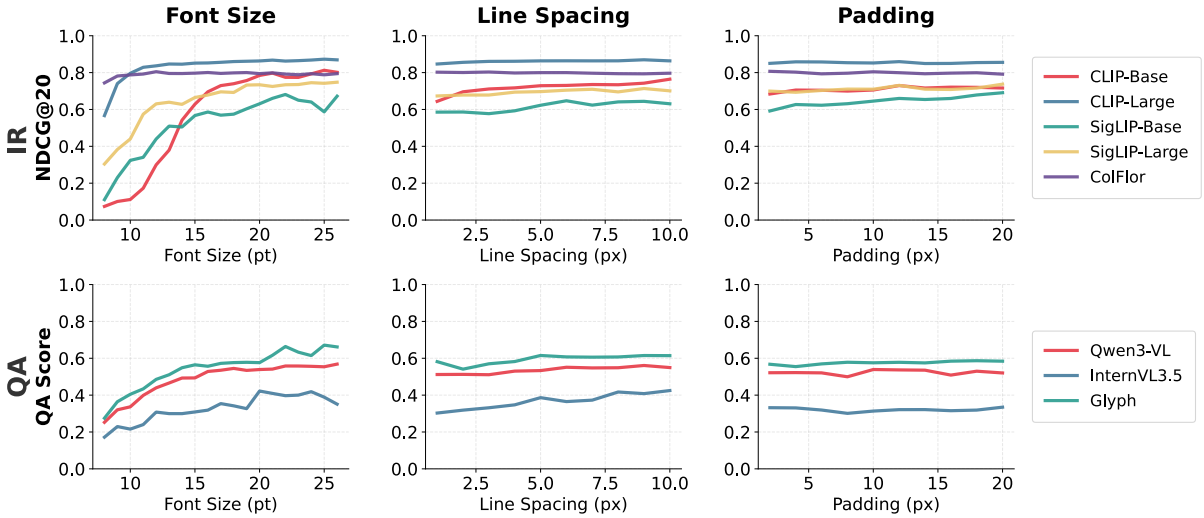


Figure 3: Effect of text density parameters on model performance. Top row: IR (nDCG@20). Bottom row: QA (avg. score). Font size has the largest impact, with different models exhibiting distinct optimal settings.

marginal $+0.6$ gain over the default (83.1), while its worst variant drops to 80.3 ($\Delta = -2.8$). Similarly, bge-L's best result is the default itself (84.6, $\Delta^* = 0.0$), with its worst variant dropping to 81.5 ($\Delta = -3.1$). The overall sensitivity ranges are narrow, and none of these text encoder differences are statistically significant.

For question answering, a more severe pattern emerges. While positive gains remain negligible: with individual models achieving maximum improvements of only $+0.5$ to $+1.8$ points over their respective defaults, improper paraphrasing causes catastrophic performance drops. The worst transformations collapse QA performance down to near floor levels ($\sim$10–12.7 points) across all models. This yields massive, statistically significant degradations (Δ) ranging from -47.1 to -54.3 points. In summary, text paraphrasing at best marginally matches the original phrasing and at worst severely disrupts LLM processing.

This stands in sharp contrast to the Visual Composition section (Table 2, top), where changing the visual layout yields statistically significant improvements (Δ^*) ranging from $+5.3$ to $+15.2$ points, with comparatively minimal downside degradation (Δ between -0.4 and -3.4). Full results across all individual transforms and models are provided in Appendix Table 20.

4.4 Key Findings

We summarize three key findings from the comparative study of input representation effects across VLMs and LLMs.

Finding 1: Visual composition significantly improves VLMs, while text paraphrasing provides no meaningful gains for LLMs. Table 2 reveals a fundamental asymmetry between visual layout formatting and textual input variations. For VLMs, strategic changes in visual composition yield statistically significant improvements (Δ^*) ranging from $+5.3$ to $+15.2$ points over default rendering. In contrast, text paraphrasing offers negligible positive returns across all evaluated LLMs, with the maximum observed gain reaching only $+1.8$ points and no improvements achieving statistical significance. This demonstrates that visual formatting represents a highly optimizable representation layer for multimodal architectures, whereas LLMs remain generally unreceptive to text-based surface-form variations as a mechanism for performance enhancement.

Finding 2: Visual composition effects are highly model-specific. Different VLMs exhibit distinct preferences for visual composition. As shown in the detailed ablation results (Appendix F), no single composition configuration performs optimally across all models. For example, CLIP-based encoders favor different density settings than SigLIP-based encoders. This model-specific sensitivity implies that visual composition must be tuned per-model rather than relying on universal heuristics.

Finding 3: Font size dominates, but no single variable suffices. Among all composition variables, font size has the largest impact on performance. The sensitivity heatmap (Appendix E) con-

firms that text density consistently ranks highest in feature importance. To isolate individual effects, we vary font size (8–26pt), line spacing (1–10px), and padding (2–20px) independently: each time holding the other two variables at fixed defaults for both visual encoders and VLMs. As shown in Figure 3, font size is the dominant factor: varying font size shifts NDCG@20 by up to 74 points for visual encoders and mean score by up to 31 points for VLMs, compared to 12 points for line spacing and 5 points for padding. Crucially, however, tuning any single variable in isolation is insufficient to reach optimal performance. Font size exhibits clear diminishing returns: most models improve steeply below 13–18pt but then plateau, and further increases can even degrade performance. In short, the composition space is high-dimensional and model-dependent: the best configuration achievable by tuning one parameter at a time is merely a lower bound on what joint optimization can reach.

These findings motivate the need for a principled optimization approach. Given the high-dimensional, model-dependent nature of the visual composition space, manual tuning is impractical. We therefore introduce **VTAOpt**, a Bayesian optimization framework that automatically discovers effective composition configurations for a given model.

5 VTAOpt: Vision-Text Augmentation Optimization

We next describe our approach for optimizing visual document composition. In the **VTAOpt** framework, we adopt same visual composition factors earlier in Table 6 and treat font size, line spacing, and padding as independent optimization variables. The full search space is detailed in Appendix B (Table 7). This design yields 24,883,200 possible configurations, making exhaustive evaluation infeasible and motivating the use of Bayesian optimization.

5.1 Objective Function

We optimize a black-box objective $J(\theta)$ over a composition configuration θ (font, layout, query type, noise, etc.). For each dataset $d \in \mathcal{D}$ we measure nDCG@20, denoted $s_d(\theta)$, on the development split. The objective is the mean score across all datasets:

$$J(\theta) = \frac{1}{|\mathcal{D}|} \sum_{d \in \mathcal{D}} s_d(\theta) \tag{1}$$

Algorithm 1 VTAOpt

Input: parameter space Θ, datasets $\mathcal{D}$, budget N, warmup n_w
Initialize history $\mathcal{H} \leftarrow \emptyset$, best score $s^* \leftarrow -\infty$
for $t = 1$ **to** N **do**
  if $t \leq n_w$ **then**
    $\theta_t \leftarrow \textsc{Sample}(\Theta)$ {Random exploration}
  else
    $\theta_t \leftarrow \text{TPE-}\textsc{Suggest}(\Theta, \mathcal{H})$ {TPE-guided}
  end if
  for $d \in \mathcal{D}$ **do**
    $\mathcal{I}_d \leftarrow \textsc{Render}(\text{corpus}_d, \theta_t)$
    $s_d \leftarrow \text{nDCG@20}(\text{queries}_d, \mathcal{I}_d)$
  end for
  $J(\theta_t) \leftarrow \frac{1}{|\mathcal{D}|} \sum_d s_d$
  if $J(\theta_t) > s^*$ **then**
    $s^* \leftarrow J(\theta_t), \quad \theta^* \leftarrow \theta_t$
  end if
  $\mathcal{H} \leftarrow \mathcal{H} \cup \{(\theta_t, J(\theta_t))\}$
end for
Return: θ^*

All datasets are evaluated in a single batched pass: documents are rendered first, then encoded together to maximize GPU utilization.

5.2 Optimization Algorithm

Bayesian Optimization with TPE. To optimize the large, discrete, and mixed-type composition space defined in Appendix B, we adopt Bayesian optimization using the Tree-structured Parzen Estimator (TPE), implemented via Optuna. Algorithm 1 summarizes the resulting optimization procedure.

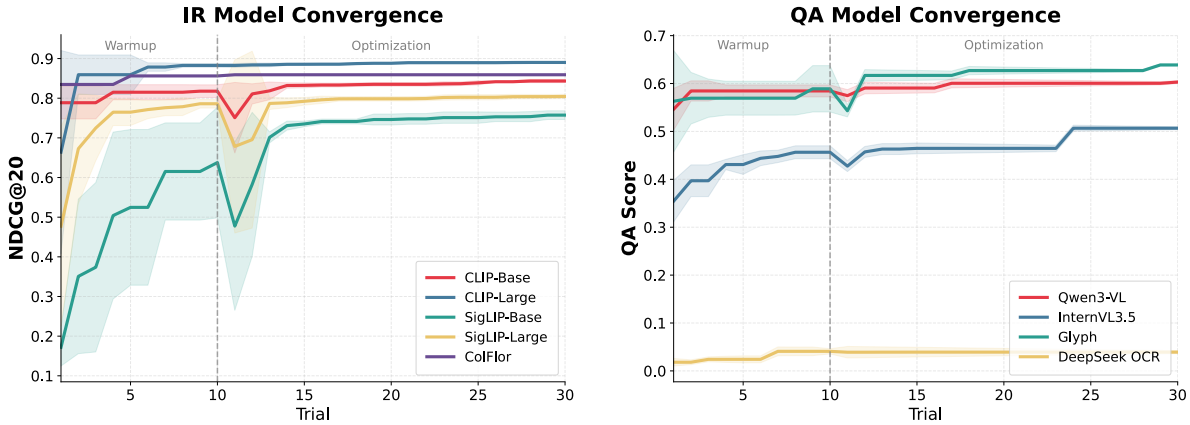
Sampling and Data Strategy. We employ a two-phase optimization protocol: a 10-trial uniform random warmup is followed by 20 trials of multivariate TPE-guided sampling to capture categorical parameter correlations via expected improvement maximization. To guarantee generalization without model tuning, we split each dataset into a 30% optimization subset for the configuration search and a held-out 70% evaluation set for final reporting.

Budget and Stopping Criteria. We enforce a strict budget of $N = 30$ trials with no early stopping. The configuration yielding the highest mean metric on the optimization subset is selected as the final output.

6 Results

6.1 VTAOpt Optimization Results

Figure 4 illustrates the efficiency of the VTAOpt search framework. While uniform random sampling during warmup yields high performance



(a) IR convergence

(b) QA convergence

Figure 4: Convergence trajectories of VTAOpt across IR and QA benchmarks. The vertical dashed line separates the exploratory random warmup phase (trials 1–10) from the Bayesian optimization phase (trials 11–30). The temporary performance dip at trial 11 occurs as the optimizer initializes its search strategy. Shaded regions denote standard deviation across 5 independent runs, showing rapid stabilization into a high-performance plateau.

Encoder	MultiReQA	NarrativeQA	Google NQ
Vision Encoders (Default)			
CLIP-B	19.5	12.1	46.4
CLIP-L	53.4	58.6	72.9
SigLIP-B	39.5	50.7	78.7
SigLIP-L	47.9	64.9	84.9
ColFlor	58.6	92.1	89.5
Vision Encoders (Random Search, 30 trials)			
CLIP-B	79.6 (+60.1)	62.5 (+50.4)	91.0 (+44.6)
CLIP-L	89.2 (+35.8)	81.3 (+22.7)	94.9 (+22.0)
SigLIP-B	60.4 (+20.9)	73.5 (+22.8)	85.5 (+6.8)
SigLIP-L	63.2 (+15.3)	66.6 (+1.7)	82.3 (-2.6)
ColFlor	62.5 (+3.9)	98.3 (+6.2)	94.1 (+4.6)
Vision Encoders (VTAOpt, 30 trials)			
CLIP-B	90.0 (+10.4)	69.7 (+7.2)	92.8 (+1.8)
CLIP-L	91.5 (+2.3)	82.1 (+0.8)	95.5 (+0.6)
SigLIP-B	65.4 (+5.0)	74.1 (+0.6)	88.3 (+2.8)
SigLIP-L	76.9 (+13.7)	75.8 (+9.2)	87.6 (+5.3)
ColFlor	64.9 (+2.4)	100.0 (+1.7)	94.9 (+0.8)

Table 3: IR results (nDCG@20, %). Parentheses show gain over Default for Random Search and gain over Random Search for VTAOpt (points).

Model	Memory	Reasoning	Retrieval
Vision–Language Models (Default)			
Qwen3-VL	38.7	22.5	92.3
Glyph	44.1	29.5	94.0
InternVL3.5	32.5	10.0	52.8
DeepSeek-OCR	5.8	0.0	2.8
Vision–Language Models (Random Search, 30 trials)			
Qwen3-VL	43.4 (+4.7)	37.0 (+14.5)	100.0 (+7.7)
Glyph	42.8 (-1.3)	46.5 (+17.0)	98.1 (+4.1)
InternVL3.5	19.4 (-13.1)	33.3 (+23.3)	81.7 (+28.9)
DeepSeek-OCR	6.4 (+0.6)	0.0 (0.0)	5.8 (+3.0)
Vision–Language Models (VTAOpt, 30 trials)			
Qwen3-VL	45.4 (+2.0)	54.3 (+17.3)	100.0 (0.0)
Glyph	48.9 (+6.1)	60.4 (+13.9)	100.0 (+2.0)
InternVL3.5	60.0 (+40.6)	33.7 (+0.4)	82.3 (+0.6)
DeepSeek-OCR	6.4 (0.0)	2.0 (+2.0)	6.0 (0.2)

Table 4: Question answering results on VTCBench (values are %×100). Memory uses Rouge-L; Reasoning/Retrieval use Contains-All. Parentheses show gain over Default for Random Search and gain over Random Search for VTAOpt (points).

variance, the transition to TPE-guided sampling drives rapid optimization. The objective stabilizes within roughly 5 additional trials, establishing near-optimal configurations by trial 15. This consistent convergence behavior across both tasks demonstrates that VTAOpt can navigate a vast search space of 24,883,200 configurations within a strict budget of 30 total trials.

Tables 3 and 4 present the full results for IR and QA, respectively, comparing three conditions: Default rendering, Random Search (best of 30 uniform samples), and VTAOpt (best of 20 TPE-guided trials plus 10 warm-up trials). *All Random Search and VTAOpt results are reported as the mean over 5 independent runs.*

Rendering Choice Matters. The gap between Default and Random Search reveals that rendering choices matter far more than commonly assumed. On IR (Table 3), sampling 30 random configurations improves CLIP-B from 26.0 to 77.7 mean nDCG@20 across datasets and lifts CLIP-L by 26.8 points on average. On QA (Table 4), Random Search boosts Qwen3-VL's reasoning from 22.5 to 37.0 (+14.5) and InternVL3.5's retrieval from

	Information Retrieval (nDCG@20)				Question Answering		
Model	**MultiReQA**	**NarrativeQA**	**Google NQ**	**Model**	**Memory**	**Reasoning**	**Retrieval**
Text Encoders				*LLMs*			
bge-L†	77.8	79.3	96.8	**Qwen3-8B**†	43.7	31.0	99.3
CLIP-L (text)	45.5	21.7	74.3	GPT-4o-mini	32.1	41.0	98.5
Vision Encoders (VTAOpt)				*VLMs (VTAOpt)*			
CLIP-L†	91.5 (+13.7)	82.1 (+2.8)	97.0 (+0.2)	Qwen3-VL	45.4 (+1.7)	54.3 (+23.3)	100.0 (+0.7)
ColFlor	64.9 (−12.9)	100.0 (+20.7)	94.9 (−1.9)	**Glyph**†	48.9 (+5.2)	60.4 (+29.4)	100.0 (+0.7)

Table 5: Comparison of text-based models with VTAOpt-optimized vision models. †Best in category. IR deltas are relative to bge-L; QA deltas are relative to Qwen3. IR is reported as nDCG@20 (%); QA Memory uses Rouge-L, Reasoning/Retrieval use Contains-All (%×100). LLMs use temperature = 0 for deterministic evaluation. VTAOpt results are the mean of 5 independent runs.

52.8 to 81.7 (+28.9). These results indicate that a large portion of vision encoder underperformance is attributable to suboptimal text rendering rather than to fundamental model limitations.

VTAOpt consistently improves over Random Search. Across all encoders and datasets, VTAOpt exceeds the best random configuration. On IR (Table 3), VTAOpt improves over Random Search for every encoder, with the largest gains for SigLIP-L (+13.7 on MultiReQA, +9.2 on NarrativeQA) and CLIP-B (+10.4 on MultiReQA). On QA (Table 4), TPE-guided search yields large reasoning gains (+17.3 for Qwen3-VL, +13.9 for Glyph) and recovers InternVL3.5's memory (+40.6) that random search had sacrificed, demonstrating its ability to balance multi-objective trade-offs.

Gains are bounded by the encoder's intrinsic capacity. DeepSeek-OCR remains near zero across all conditions (Table 4), confirming that composition optimization cannot compensate for models that fundamentally lack the ability to process rendered text. VTAOpt unlocks latent model capability through better rendering, but it cannot create capability that does not exist. In all cases, VTAOpt finds competitive configurations within 30 trials—exploring fewer than 0.00013% of the full 24,883,200-configuration space.

6.2 Effectiveness of Visual Composition Optimization

IR. When evaluating text retrieval, optimizing the visual form of text via VTAOpt allows vision encoders to consistently match or exceed established text-native baselines (Table 5 left side). Specifically, CLIP-L with VTAOpt surpasses bge-L (the strongest text encoder baseline) on MultiReQA by a substantial margins of 13.7 points (91.5 vs. 77.8) and on NarrativeQA (82.1 vs. 79.3), while achieving comparable performance on Google NQ (97.0 vs. 96.8). It is particularly striking given that CLIP-L's native text encoder scores only 45.5 and 21.7 on MultiReQA and NarrativeQA. By rendering text as images and optimizing layout factors, VTAOpt elevates a standard vision encoder beyond highly specialized text models. Similarly, ColFlor thrives within this optimized space, surpassing CLIP-L's text encoder on all three datasets.

QA. On question answering tasks, optimizing the visual layout via VTAOpt enables vision-language models to outperform text baselines (Table 5 right side). Specifically, Glyph demonstrates notable efficacy across all three subtasks, consistently exceeding Qwen3-8B and notably outpacing it on reasoning by a substantial 29.4 points. Qwen3-VL similarly showcases the strength of this visual optimization axis, yielding substantial gains on reasoning and achieving ceiling performance on retrieval. Crucially, as detailed in Appendix Table 20, both optimized VLMs match or exceed the performance of their corresponding backbone LLMs. Both systems also consistently outpace GPT-4o-mini, indicating the strong effectiveness of visual composition.

Implications. These findings demonstrate that the visual rendering space affords VLMs a higher performance ceiling than the text space affords LLMs. By framing visual layout as an optimization axis, VTAOpt unlocks latent capabilities without fine-tuning. This establishes a new frontier for NLP optimization beyond traditional prompt engineering, proving that visual text presentation is just as critical as the text itself.

Limitations and Future Work

Currently, VTAOpt is engineered exclusively to maximize downstream task performance to demonstrate the raw efficacy of the visual composition axis. However, an ideal production-ready framework must balance this performance against operational efficiency, specifically optimizing the trade-off between visual compression ratios (token length) and task accuracy. Future iterations will incorporate multi-objective optimization to address these computational overhead constraints.

Since Visual Text Composition (VTC) is an emerging paradigm, existing vision-language models are generalists; there are currently no vision encoders explicitly pre-trained for targeted tasks, such as a visual equivalent of a text-based sentence transformer optimized for information retrieval. Consequently, establishing a perfectly baseline-to-baseline fair comparison remains difficult. This work focuses on demonstrating the latent potential of the visual axis using existing architectures, paving the way for future task-specific visual pre-training.

References

Joshua Ainslie, James Lee-Thorp, Michiel De Jong, Yury Zemlyanskiy, Federico Lebrón, and Sumit Sanghai. 2023. Gqa: Training generalized multi-query transformer models from multi-head checkpoints. *arXiv preprint arXiv:2305.13245*.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xiong-Hui Chen, Zesen Cheng, and 1 others. 2025. Qwen3-vl technical report. *ArXiv*, abs/2511.21631.

Jiale Cheng, Yusen Liu, Xinyu Zhang, Yulin Fei, Wenyi Hong, Ruiliang Lyu, Weihan Wang, Zhe Su, Xiaotao Gu, Xiao Liu, and 1 others. 2025. Glyph: Scaling context windows via visual-text compression. *arXiv preprint arXiv:2510.17800*.

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. 2019. Generating long sequences with sparse transformers. *Preprint*, arXiv:1904.10509.

Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in neural information processing systems*, 35:16344–16359.

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

Mandy Guo, Yinfei Yang, Daniel Cer, Qinlan Shen, and Noah Constant. 2021. MultiReQA: A cross-domain evaluation forRetrieval question answering models. In *Proceedings of the Second Workshop on Domain Adaptation for NLP*, pages 94–104, Kyiv, Ukraine. Association for Computational Linguistics.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Tomáš Kočiský, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2018. The NarrativeQA reading comprehension challenge. *Transactions of the Association for Computational Linguistics*, 6:317–328.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR.

Yanhong Li, Zixuan Lan, and Jiawei Zhou. 2025. Text or pixels? it takes half: On the token efficiency of visual text inputs in multimodal llms. *arXiv preprint arXiv:2510.18279*.

Ahmed Masry. 2024. Colflor: Towards bert-size vision-language document retrieval models. Hugging Face Blog. Model: `https://huggingface.co/ahmed-masry/ColFlor`.

Ofir Press, Noah A Smith, and Mike Lewis. 2021. Train short, test long: Attention with linear biases enables input length extrapolation. *arXiv preprint arXiv:2108.12409*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. *Preprint*, arXiv:2103.00020. Checkpoint: `https://huggingface.co/openai/clip-vit-large-patch14`.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics. Checkpoint: `https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2`.

Noam Shazeer. 2019. Fast transformer decoding: One write-head is all you need. *arXiv preprint arXiv:1911.02150*.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. 2024. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063.

Maria Tsimpoukelli, Jacob L Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill. 2021. Multimodal few-shot learning with frozen language models. *Advances in Neural Information Processing Systems*, 34:200–212.

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

Alex Jinpeng Wang, Linjie Li, Yiqi Lin, Min Li, Lijuan Wang, and Mike Zheng Shou. 2024. Leveraging visual tokens for extended text contexts in multi-modal learning. *Advances in Neural Information Processing Systems*, 37:14325–14348.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, and 1 others. 2025. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*.

Yibo Wang, Yongcheng Jing, Shunyu Liu, Hao Guan, Rong-cheng Tu, Chengyu Wang, Jun Huang, and Dacheng Tao. 2026. Vtc-r1: Vision-text compression for efficient long-context reasoning. *arXiv preprint arXiv:2601.22069*.

Haoran Wei, Yaofeng Sun, and Yukun Li. 2025. Deepseek-ocr: Contexts optical compression. *arXiv preprint arXiv:2510.18234*.

Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. 2024. C-pack: Packed resources for general chinese embeddings. In *Proceedings of the 47th international ACM SIGIR conference on research and development in information retrieval*, pages 641–649.

Ling Xing, Alex Jinpeng Wang, Rui Yan, Xiangbo Shu, and Jinhui Tang. 2025. Vision-centric token compression in large language model. *arXiv preprint arXiv:2502.00791*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. 2023. Sigmoid loss for language image pre-training. *Preprint*, arXiv:2303.15343. Checkpoint: https://huggingface.co/google/siglip-base-patch16-224.

Xin Zhang, Yanzhao Zhang, Dingkun Long, Wen Xie, Ziqi Dai, Jialong Tang, Huan Lin, Baosong Yang, Pengjun Xie, Fei Huang, and 1 others. 2024. mgte: Generalized long-context text representation and reranking models for multilingual text retrieval. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 1393–1412.

Hongbo Zhao, Meng Wang, Fei Zhu, Wenzhuo Liu, Bolin Ni, Fanhu Zeng, Gaofeng Meng, and Zhaoxiang Zhang. 2025. Vtcbench: Can vision-language models understand long context with vision-text compression? *arXiv preprint arXiv:2512.15649*.

A Visual Document Composition Factors

Category	Options	# Settings
TEXT DENSITY		
High	font=16, spacing=2, padding=1	1
Medium	font=18, spacing=4, padding=10	1
Low	font=22, spacing=8, padding=20	1
TYPOGRAPHY		
Font family	{DejaVu, Times, Courier}	3
Font style	{regular, bold, italic, underline}	4
NOISE		
Noise mode	{none, gaussian, poisson, uniform}	4
LAYOUT		
Columns	{1, 2}	2
Paragraph mode	{default, sep, cue, sep+cue}	4
CONTEXT-AWARE HIGHLIGHTING		
Highlight mode	{none, nouns, verbs, topic}	4
COLOR SCHEME		
Color palette	9 predefined schemes	9
QUERY		
Query type	{text, image}	2
Total		**35**

Table 6: Visual document composition factors used in our systematic study

B VTAOpt Search Space

Parameter	Range	#
TEXT DENSITY AND TYPOGRAPHY		
Text size	8–26 px	10
Line spacing	1–10 px	10
Padding	1–20 px	9
Font family	{DejaVu, Times, Courier}	3
Font style	{regular, bold, italic, underline}	4
NOISE AND VISUAL PERTURBATIONS		
Noise mode	{none, gaussian, poisson, uniform}	4
LAYOUT		
Columns	{1, 2}	2
Paragraph mode	{default, sep, cue, sep+cue}	4
CONTEXT-AWARE HIGHLIGHTING		
Highlight mode	{none, nouns, verbs, topic}	4
COLOR SCHEME		
Color palette	9 predefined schemes	9
QUERY		
Query type	{text, image}	2

Table 7: VTAOpt search space (24.88M configurations).

C Model Information

Model Type	Model Name	Provider	Params	HF ID
Information Retrieval				
Vision encoder	CLIP-B (Radford et al., 2021)	OpenAI	151M	`openai/clip-vit-base-patch32`
Vision encoder	CLIP-L (Radford et al., 2021)	OpenAI	304M	`openai/clip-vit-large-patch14`
Vision encoder	SigLIP-B (Zhai et al., 2023)	Google	203M	`google/siglip-base-patch16-224`
Vision encoder	SigLIP-L (Zhai et al., 2023)	Google	652M	`google/siglip-large-patch16-256`
Vision encoder	ColFlor (Masry, 2024)	ColPali	174M	`ahmed-masry/ColFlor`
Text encoder	MiniLM (Reimers and Gurevych, 2019)	UKP Lab	22M	`sentence-transformers/all-MiniLM-L6-v2`
Text encoder	RoBERTa-B	Meta	125M	`roberta~\cite{liu2019roberta}`
Text encoder	CLIP-L-Text (Radford et al., 2021)	OpenAI	123M	`openai/clip-vit-large-patch14`
Text encoder	bge-L (Xiao et al., 2024)	BAAI	335M	`BAAI/bge-large-en-v1.5`
Text encoder	gte-L (Zhang et al., 2024)	Alibaba-NLP	434M	`Alibaba-NLP/gte-large-en-v1.5`
Question Answering				
Vision–language model	Qwen3-VL (Bai et al., 2025)	Alibaba (Qwen)	8B	`Qwen/Qwen3-VL-8B-Instruct`
Vision–language model	DeepSeek-OCR (Wei et al., 2025)	DeepSeek	3B	`deepseek-ai/DeepSeek-OCR`
Vision–language model	InternVL3.5 (Wang et al., 2025)	OpenGVLab	8B	`OpenGVLab/InternVL3_5-8B-HF`
Vision–language model	Glyph (Cheng et al., 2025)	ZAI	9B	`zai-org/Glyph`
Large language model	Qwen3 (Yang et al., 2025)	Alibaba	8B	`Qwen/Qwen3-8B`
Large language model	Llama-8B (Grattafiori et al., 2024)	Meta	8B	`meta-llama/Meta-Llama-3-8B`
Large language model	GPT-4o-mini (Hurst et al., 2024)	OpenAI	–	API-based / `gpt-4o-mini`

Table 8: Model inventory used in our experiments.

D Composition Variables

D.1 Text Density

Text density controls how much textual content can be rendered within a fixed image canvas and is a primary factor governing compression under a bounded visual token budget. We parameterize text density through three rendering variables: *font size*, *line spacing* (the vertical gap between adjacent lines), and *page padding*, which determines the margins around the text.

For controlled analysis of density effects, we define three representative density regimes: *high* density (font size 16, line spacing 2, padding 1), *mid* density (font size 18, line spacing 4, padding 10), and *low* density (font size 22, line spacing 8, padding 20). These settings are chosen to span a broad range of content coverage and readability, enabling us to isolate the impact of text density on retrieval performance.

D.2 Typography

We study typographic effects using a controlled rendering setup where text density is held constant (mid density: font size 18pt, line spacing 4px, padding 10px). We then vary two typography-related parameters: *font family* {sans-serif (DejaVu Sans), serif (Apple Times), monospace (Courier New)}, *font style* {regular, normal, bold, italic}. To avoid confounding typography with content truncation, we render each document into multiple pages when needed so that all text is preserved under every typographic condition. We evaluate downstream performance across these settings to quantify how glyph shape and character-level spacing influence encoder performance on long-document tasks.

D.3 Noise and Visual Perturbations

We examine the effect of visual noise as a composition variable that perturbs the rendered document while preserving its textual content. We inject pixel-level noise into the rendered images using three

noise models: *Gaussian*, *Poisson*, and *Uniform* noise. Noise is applied uniformly across the image using parameterizations defined by the corresponding noise models, without altering document layout, typography, or text density. By isolating noise from other composition factors, we quantify how visual perturbations interact with document structure and affect retrieval performance under the document-as-image representation.

D.4 Color & Luminance Design

We study color and luminance effects using a controlled rendering setup consistent with the typography ablation. Text density is held on mid density (font size 18pt, line spacing 4px, padding 10px), with all typographic parameters fixed (DejaVu Sans, regular weight, normal style, normal letter spacing). We construct nine foreground–background configurations across three contrast levels: high (15–21:1), mid (10–14:1), and low (2.0–2.5:1). These configurations are illustrated in Figure 2: high contrast (p–r), mid contrast (s–u), and low contrast (v–x). Each level includes three configurations: neutral or same-family warm colors (light background), same-family cool colors (light or dark background), and complementary color pairs (light or dark background). This systematic design varies four factors—contrast ratio, background polarity, color temperature, and color harmony—with balanced representation of light (6) and dark (3) backgrounds across all contrast levels. Documents are rendered into multiple pages when necessary to avoid content truncation.

D.5 Layout and Spatial Structure

We evaluate layout and spatial composition under a controlled rendering setup where page resolution is fixed to 512×512 and typography is held constant (font family, font size, line spacing, and padding fixed to the default setting). We vary two layout factors that change reading order and paragraph structure: *columns* $\{1, 2\}$, *paragraph separation* {compact (no extra gap), spaced (+8px between paragraphs)}, and a *paragraph cue* applied at paragraph boundaries (none vs. an enlarged first word).

D.6 Content-aware Highlighting

We study whether vision encoders benefit from explicit visual cues by introducing content-aware highlighting strategies that modify the appearance of selected words while keeping the underlying text unchanged. We highlight tokens according to four selection rules: (1) *part-of-speech highlighting*, where nouns or verbs are emphasized; and (2) *topic highlighting*, where the top-ranked topic terms extracted from the context are emphasized. Highlighting is implemented using consistent visual operations (e.g., boldface and underlining) applied only to the selected tokens. We compare these strategies against a no-highlighting baseline to quantify whether content-aware emphasis improves retrieval and question answering performance in the document-as-image setting.

D.7 Query Type

We also examine the effect of *query type* on downstream performance. In our setting, queries can be provided either as raw text or rendered as images, and both formats are supported by the vision encoders and vision–language models evaluated in this work. Although query type is not a document composition factor, we include it here because we find it has a large impact on performance across both information retrieval and question answering tasks, as illustrated in Table 9. In practice, different models exhibit markedly different preferences for text-based versus image-based queries, leading to substantial performance differences under otherwise identical settings.

Information Retrieval *Metric: nDCG@20*						
Model	**MultiReQA**		**NarrativeQA**		**Google NQ**	
	img-q	**txt-q**	**img-q**	**txt-q**	**img-q**	**txt-q**
CLIP-B	19.5	**60.2**	12.1	**52.3**	46.4	**87.6**
SigLIP-B	**39.5**	11.7	**50.7**	10.4	**78.7**	33.5
Question Answering *Memory: ROUGE-L; Reasoning/Retrieval: Contains-All*						
Model	**Memory**		**Reasoning**		**Retrieval**	
	img-q	**txt-q**	**img-q**	**txt-q**	**img-q**	**txt-q**
Qwen3-VL	33.2	**38.7**	22.0	**22.5**	90.3	**92.3**
InternVL3.5	28.4	**32.5**	8.5	**10.0**	46.0	**52.8**

Table 9: Img-q vs txt-q comparison (values are % $\times 100$). Within each (img-q, txt-q) pair, the better value is in **bold**.

E Sensitivity Heatmap of Visual Composition Effects

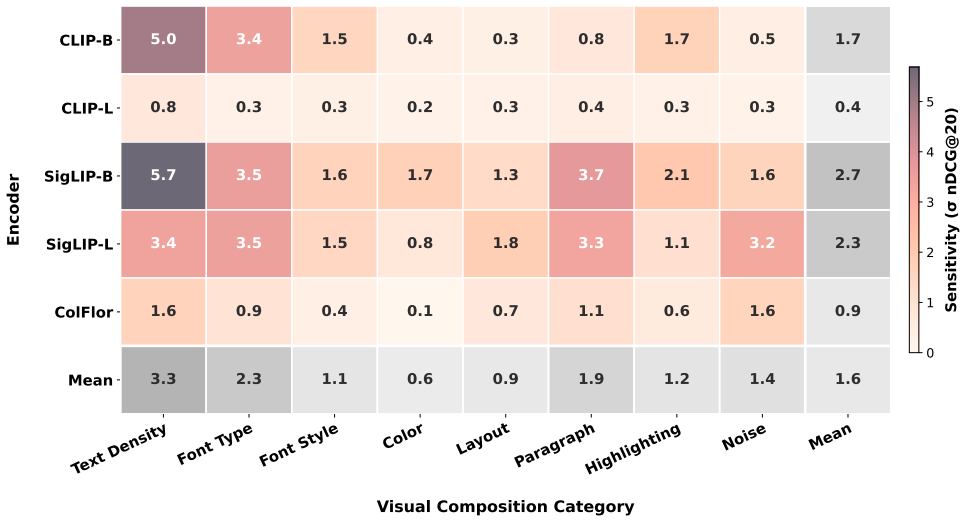


Figure 5: Sensitivity heatmap of visual composition effects. Each cell reports the standard deviation of nDCG@20 across variants within a composition category, averaged across datasets, for a given encoder. Marginal means across encoders and composition categories are shown on the right and bottom.

F Detailed Ablation Results for Visual Composition Factors

F.1 Text Density Ablation Results

Dataset	Metric	CLIP-base			CLIP-large			SigLIP-base			SigLIP-large		
		High	**Mid**	**Low**	**High**	**Mid**	**Low**	**High**	**Mid**	**Low**	**High**	**Mid**	**Low**
MultiReQA	R@1	34.2	39.1	**44.1**	47.8	48.5	48.5	16.2	19.3	**24.2**	23.6	27.3	**30.4**
	R@5	70.2	88.8	**92.6**	93.2	98.1	98.1	44.7	60.3	**66.5**	63.4	64.6	**71.4**
	R@10	84.5	95.7	95.7	100.0	100.0	100.0	59.0	72.7	**77.0**	67.7	75.8	**83.9**
	R@20	91.3	96.3	**99.4**	100.0	100.0	100.0	73.3	74.5	**87.6**	74.5	**86.3**	85.7
	MRR@20	50.8	59.4	**63.7**	67.0	**68.6**	68.3	29.7	36.2	**41.7**	39.6	43.6	**47.8**
	MAP	51.0	59.5	**63.7**	67.0	**68.6**	68.3	30.3	36.9	**42.1**	40.2	43.9	**48.2**
	nDCG@20	60.2	68.4	**72.4**	75.2	**76.5**	76.3	39.5	45.4	**52.4**	47.9	53.5	**56.9**
NarrativeQA	R@1	37.1	49.5	**60.4**	68.1	71.1	**72.7**	36.8	51.1	**63.0**	52.6	61.9	**67.9**
	R@5	57.4	68.2	**77.3**	82.7	84.8	**86.3**	55.1	69.1	**77.2**	69.5	76.7	**81.6**
	R@10	64.9	74.5	**82.5**	87.0	89.0	**90.0**	62.2	75.1	**81.7**	75.3	81.5	**85.7**
	R@20	72.4	80.4	**87.0**	90.7	92.3	**93.1**	69.6	80.7	**85.8**	80.6	85.7	**89.2**
	MRR@20	46.4	57.9	**68.0**	74.7	77.3	**78.7**	45.2	59.2	**69.4**	60.2	68.6	**74.1**
	MAP	46.8	58.2	**68.2**	74.8	77.5	**78.9**	45.6	59.5	**69.6**	60.5	68.8	**74.3**
	nDCG@20	52.3	63.1	**72.4**	78.4	80.8	**82.1**	50.7	64.2	**73.2**	64.9	72.5	**77.6**
Google NQ	R@1	79.3	83.0	**84.3**	91.3	91.3	**92.0**	67.0	71.7	**76.0**	76.0	78.3	**79.3**
	R@5	91.3	93.3	**95.3**	96.0	**96.7**	96.3	83.7	85.3	**87.7**	88.0	90.3	**92.0**
	R@10	94.0	95.0	**97.3**	98.3	98.3	98.3	89.3	90.3	**92.7**	92.7	93.3	**95.0**
	R@20	96.0	97.0	**97.7**	99.0	98.7	99.0	91.7	93.0	**95.7**	95.0	96.3	**97.0**
	MRR@20	85.0	87.8	**89.4**	93.7	93.9	**94.2**	74.8	78.3	**81.6**	81.9	83.6	**85.1**
	MAP	85.0	87.9	**89.4**	93.8	94.0	**94.3**	74.9	78.5	**81.7**	82.0	83.7	**85.1**
	nDCG@20	87.6	90.0	**91.4**	95.0	95.1	**95.4**	78.7	81.7	**84.9**	84.9	86.6	**87.9**

Table 10: Text density ablation (3 levels). All metrics are reported as **percentages** ($\times 100$; higher is better), i.e., R@K, MRR@20, MAP, and nDCG@20 are shown on a 0–100 scale.

Table 11: Text density ablation (3 levels) on VTCBench question answering. All metrics are reported as **percentages** ($\times 100$; higher is better). **Memory** uses **Rouge-L**; **Reasoning** and **Retrieval** use **Contains-All**. Density mapping: High = (font=16, spacing=2, padding=1); Mid = (font=18, spacing=4, padding=10); Low = (font=22, spacing=8, padding=20).

Dataset	Metric	Qwen3-VL-8B			DeepSeek-OCR			InternVL3.5-8B			Glyph		
		High	**Mid**	**Low**	**High**	**Mid**	**Low**	**High**	**Mid**	**Low**	**High**	**Mid**	**Low**
Memory	Rouge-L	38.7	41.4	**42.8**	**5.8**	3.0	3.2	32.5	**34.5**	18.3	44.1	48.8	48.6
Reasoning	Contains-All	22.5	**28.0**	25.5	0.0	0.0	0.0	10.0	15.0	**16.0**	29.5	35.5	**50.6**
Retrieval	Contains-All	92.3	**98.3**	91.3	2.8	**3.5**	2.7	52.8	**72.8**	51.2	94.0	**95.0**	94.1

F.2 Font-type Ablation Results

Dataset	Metric	CLIP base			CLIP large			SigLIP-base			SigLIP-large		
		DejaVu	Times	Courier	DejaVu	Times	Courier	DejaVu	Times	Courier	DejaVu	Times	Courier
MultiReQA	R@1	**39.1**	29.2	31.7	48.5	49.7	49.7	19.3	15.5	**21.7**	27.3	20.5	**28.6**
	R@5	**88.8**	76.4	75.2	**98.1**	95.7	96.9	**60.3**	38.5	54.7	64.6	52.2	**70.2**
	R@10	**95.7**	84.5	87.6	100.0	99.4	100.0	**72.7**	49.1	64.0	75.8	66.5	**78.9**
	R@20	**96.3**	88.8	89.4	100.0	100.0	100.0	**74.5**	62.1	71.4	86.3	72.7	**89.4**
	MRR@20	**59.4**	48.4	50.2	68.6	68.9	68.9	**36.2**	26.8	35.4	43.6	34.7	**46.1**
	MAP	**59.5**	48.7	50.5	68.6	68.9	68.9	**36.9**	27.3	36.1	43.9	35.4	**46.4**
	nDCG@20	**68.4**	58.1	59.6	76.5	76.6	76.6	**45.4**	34.7	43.8	53.5	43.5	**56.2**
NarrativeQA	R@1	**49.5**	38.3	38.4	**71.1**	69.4	69.0	**51.1**	47.1	45.9	61.9	57.5	**63.7**
	R@5	**68.2**	57.7	58.0	**84.8**	83.9	84.1	**69.1**	65.1	63.3	76.7	73.8	**78.7**
	R@10	**74.5**	65.8	65.9	**89.0**	88.2	88.4	**75.1**	71.3	69.5	81.5	79.0	**83.1**
	R@20	**80.4**	73.6	73.4	**92.3**	91.8	92.1	**80.7**	77.5	76.0	85.7	83.7	**87.3**
	MRR@20	**57.9**	47.2	47.4	**77.3**	76.0	75.8	**59.2**	55.3	53.7	68.6	64.8	**70.5**
	MAP	**58.2**	47.6	47.8	**77.5**	76.1	75.9	**59.5**	55.6	54.0	68.8	65.0	**70.7**
	nDCG@20	**63.1**	53.2	53.3	**80.8**	79.6	79.6	**64.2**	60.4	58.8	72.5	69.2	**74.4**
Google NQ	R@1	**83.0**	77.0	82.3	**91.3**	89.7	90.3	**71.7**	62.3	69.7	78.3	70.3	78.3
	R@5	93.3	93.3	91.3	96.7	**97.0**	96.7	**85.3**	76.7	84.7	90.3	85.0	**91.7**
	R@10	95.0	**95.3**	94.7	98.3	98.0	98.3	**90.3**	82.0	87.0	93.3	88.7	**93.7**
	R@20	**97.0**	96.7	96.0	98.7	98.7	**99.0**	**93.0**	86.7	92.0	96.3	92.7	96.3
	MRR@20	**87.8**	84.4	86.8	**93.9**	92.9	93.3	**78.3**	69.2	76.4	83.6	77.1	**83.8**
	MAP	**87.9**	84.5	86.9	**94.0**	92.9	93.4	**78.5**	69.4	76.5	83.7	77.3	**83.9**
	nDCG@20	**90.0**	87.4	89.0	**95.1**	94.3	94.7	**81.7**	73.2	80.0	86.6	80.7	**86.8**

Table 12: Font-type ablation (3 fonts). Scores are reported as $\times 100$ (higher is better). Fonts: DejaVu Sans, Times, Courier New.

Table 13: Font-type ablation (3 fonts) on VTCBench question answering. All metrics are reported as **percentages** ($\times 100$; higher is better). **Memory** uses **Rouge-L**; **Reasoning** and **Retrieval** use **Contains-All**. Fonts: DejaVu Sans, Times, Courier New.

Dataset	Metric	Qwen3-VL-8B			DeepSeek-OCR			InternVL3.5-8B			Glyph		
		DejaVu	**Times**	**Courier**	**DejaVu**	**Times**	**Courier**	**DejaVu**	**Times**	**Courier**	**DejaVu**	**Times**	**Courier**
Memory	Rouge-L	38.7	37.3	**40.6**	**5.8**	0.1	2.9	32.5	**33.6**	28.8	**44.1**	41.9	43.1
Reasoning	Contains-All	22.5	21.0	**23.0**	0.0	0.0	0.0	10.0	9.0	**10.5**	**29.5**	24.0	25.2
Retrieval	Contains-All	**92.3**	91.8	91.5	**2.8**	0.0	0.0	**52.8**	51.2	48.5	94.0	**94.3**	93.8

F.3 Font-style Ablation Results

Dataset	Metric	CLIP-base				CLIP-large				SigLIP-base				SigLIP-large			
		Reg.	Bold	Ital.	Und.	Reg.	Bold	Ital.	Und.	Reg.	Bold	Ital.	Und.	Reg.	Bold	Ital.	Und.
MultiReQA	R@1	39.1	**40.4**	39.8	37.3	48.4	49.1	**50.3**	49.1	19.2	21.1	**22.4**	20.5	27.3	26.1	**31.1**	27.3
	R@5	88.8	84.5	88.8	82.6	98.1	95.7	97.5	98.1	**60.2**	50.3	59.6	52.8	64.6	63.3	**73.9**	65.2
	R@10	**95.7**	93.2	95.0	90.1	100.0	100.0	100.0	100.0	**72.7**	64.6	69.6	67.1	75.8	68.3	**83.9**	76.4
	R@20	96.3	95.7	**99.4**	96.3	100.0	100.0	100.0	100.0	74.5	77.0	**77.6**	71.4	86.3	86.3	84.5	**91.9**
	MRR@20	59.4	59.2	**59.7**	55.6	68.6	68.4	**69.7**	68.5	36.2	34.9	**38.9**	34.9	43.6	42.0	**49.3**	43.8
	MAP	59.5	59.4	**59.7**	55.7	68.6	68.4	**69.7**	68.5	36.9	35.6	**39.4**	35.6	43.9	42.3	**49.7**	43.9
	nDCG@20	68.4	68.1	**69.3**	65.3	76.4	76.3	**77.3**	76.4	45.4	44.5	**48.0**	43.4	53.5	52.1	**57.9**	54.7
NarrativeQA	R@1	**49.5**	49.4	49.2	43.4	71.1	**71.7**	71.4	71.3	51.1	**59.0**	52.1	50.4	61.9	59.9	**63.5**	60.6
	R@5	**68.2**	67.5	67.4	62.5	84.8	**85.4**	85.1	85.0	69.1	**74.4**	69.6	67.9	76.7	75.4	**78.3**	75.9
	R@10	**74.5**	73.9	73.9	69.9	89.0	**89.5**	89.2	88.9	75.1	**79.1**	75.5	74.0	81.4	80.2	**83.0**	80.8
	R@20	**80.4**	80.1	80.3	77.0	92.3	**92.7**	92.3	92.0	80.7	**83.7**	80.9	79.7	85.7	84.6	**87.3**	85.2
	MRR@20	**57.9**	57.6	57.6	52.1	77.3	**77.9**	77.6	77.4	59.2	**66.0**	60.0	58.4	68.6	67.0	**70.2**	67.6
	MAP	**58.2**	57.9	57.9	52.5	77.5	**78.0**	77.8	77.5	59.5	**66.2**	60.3	58.7	68.8	67.2	**70.4**	67.8
	nDCG@20	**63.0**	62.7	62.8	57.8	80.8	**81.3**	81.1	80.8	64.1	**70.1**	64.8	63.3	72.5	71.0	**74.2**	71.6
Google NQ	R@1	83.0	**86.3**	84.3	82.3	91.3	91.3	**91.7**	90.7	71.7	72.0	73.3	70.7	78.3	76.3	77.0	**81.0**
	R@5	93.3	**94.3**	93.7	93.3	96.7	**97.0**	96.7	96.3	85.3	**86.3**	83.3	84.3	90.3	90.0	89.3	**91.0**
	R@10	95.0	**96.3**	95.3	94.7	98.3	98.3	98.3	97.7	**90.3**	90.0	90.0	88.0	93.3	93.3	92.3	**93.7**
	R@20	97.0	**97.7**	97.0	96.3	98.7	**99.3**	98.7	98.7	93.0	93.0	**94.0**	93.0	96.3	94.7	96.7	**97.3**
	MRR@20	87.8	**89.9**	88.4	87.2	93.9	94.0	**94.3**	93.1	**78.3**	77.9	78.3	76.8	83.6	82.5	82.8	**85.6**
	MAP	87.9	**90.0**	88.4	87.3	94.0	94.0	**94.3**	93.1	**78.5**	78.1	78.4	76.9	83.7	82.7	82.8	**85.7**
	nDCG@20	90.0	**91.8**	90.4	89.4	95.1	95.3	**95.4**	94.4	81.7	81.5	**81.8**	80.5	86.6	85.5	86.0	**88.4**

Table 14: Font-type ablation (3 fonts) on VTCBench question answering. All metrics are reported as **percentages** ($\times 100$; higher is better). **Memory** uses **Rouge-L**; **Reasoning** and **Retrieval** use **Contains-All**. Fonts: DejaVu Sans, Times, Courier New.

F.4 Column Layout Ablation Results

F.5 Column Layout Ablation Results

Dataset	Metric	CLIP-B		CLIP-L		SigLIP-B		SigLIP-large	
		1C	2C	1C	2C	1C	2C	1C	2C
MultiReQA	R@1	39.1	**39.8**	48.5	**50.3**	19.3	**23.0**	**27.3**	24.8
	R@5	**88.8**	86.9	**98.1**	93.8	**60.3**	59.6	**64.6**	61.5
	R@10	**95.7**	95.0	100.0	100.0	**72.7**	68.3	75.8	**76.4**
	R@20	96.3	**99.4**	100.0	100.0	74.5	**77.6**	**86.3**	84.5
	MRR@20	**59.4**	58.8	68.6	**69.3**	36.2	**37.5**	**43.6**	41.8
	MAP@100	**59.5**	58.9	68.6	**69.3**	36.9	**38.0**	**43.9**	42.1
	nDCG@20	68.4	**68.6**	76.5	**76.9**	45.4	**46.8**	**53.5**	51.7
NarrativeQA	R@1	49.5	**50.9**	71.1	**71.4**	**51.1**	45.5	**61.9**	53.7
	R@5	68.2	**69.6**	84.8	**85.1**	**69.1**	62.3	**76.7**	70.6
	R@10	74.5	**76.2**	89.0	**89.1**	**75.1**	68.2	**81.5**	76.3
	R@20	80.4	**81.8**	92.3	**92.4**	**80.7**	74.7	**85.7**	81.7
	MRR@20	57.9	**59.5**	77.3	**77.6**	**59.2**	53.1	**68.6**	61.3
	MAP@100	58.2	**59.8**	77.5	**77.7**	**59.5**	53.5	**68.8**	61.6
	nDCG@20	63.1	**64.6**	80.8	**81.0**	**64.2**	58.0	**72.5**	66.0
Google NQ	R@1	**83.0**	82.3	**91.3**	88.3	**71.7**	72.0	**78.3**	76.0
	R@5	**93.3**	92.0	96.7	97.0	**85.3**	85.0	**90.3**	86.3
	R@10	95.0	**95.7**	98.3	98.7	90.3	**90.7**	**93.3**	91.0
	R@20	97.0	97.0	98.7	**99.3**	**93.0**	94.3	**96.3**	93.7
	MRR@20	**87.8**	87.2	**93.9**	92.2	**78.3**	77.7	**83.6**	81.2
	MAP@100	**87.9**	87.3	**94.0**	92.2	**78.5**	77.8	**83.7**	81.3
	nDCG@20	**90.0**	89.6	**95.1**	93.9	**81.7**	81.5	**86.6**	84.1

Table 15: One-column (1C) vs two-column (2C) layout comparison. All metrics are reported as $\times 100$ (higher is better). Best result between 1C and 2C is highlighted in bold.

F.6 Paragraph Layout Ablation Results

Dataset	Metric	CLIP-B				CLIP-L				SigLIP-B				SigLIP-L			
		Def.	**Sep.**	**Cue**	**S+C**	**Def.**	**Sep.**	**Cue**	**S+C**	**Def.**	**Sep.**	**Cue**	**S+C**	**Def.**	**Sep.**	**Cue**	**S+C**
NarrativeQA	R@1	49.5	50.5	52.5	**53.3**	71.1	**71.6**	70.0	70.7	51.1	**52.1**	40.4	39.9	**61.9**	60.4	54.4	53.0
	R@5	68.2	69.3	**71.1**	70.5	84.8	**85.2**	84.7	84.9	**69.1**	68.4	56.9	56.8	**76.7**	75.8	70.5	69.2
	R@10	74.5	75.8	**77.1**	76.4	89.0	**89.2**	88.8	88.7	**75.1**	73.9	63.2	63.5	**81.5**	80.6	75.5	74.8
	R@20	80.4	81.8	**82.5**	81.7	92.3	**92.5**	92.0	92.0	**80.7**	79.5	69.7	69.9	**85.7**	84.9	80.3	80.1
	MRR@20	57.9	59.0	60.8	**61.0**	77.3	**77.8**	76.6	77.1	59.2	**59.5**	47.9	47.6	**68.6**	67.4	61.6	60.4
	MAP@100	58.2	59.3	61.1	**61.3**	77.5	**77.9**	76.7	77.2	59.5	**59.8**	48.3	48.0	**68.8**	67.6	61.9	60.6
	nDCG@20	63.0	64.3	**65.8**	**65.8**	80.8	**81.2**	80.2	80.6	**64.1**	**64.1**	52.9	52.7	**72.5**	71.4	65.9	64.9
GoogleNQ	R@1	83.0	**84.0**	83.0	83.3	**91.3**	90.3	89.3	88.7	71.7	**73.3**	65.3	69.0	78.3	**78.7**	71.7	68.7
	R@5	**93.3**	**93.3**	92.3	91.3	**96.7**	96.3	95.7	95.7	85.3	**89.0**	84.7	87.0	**90.3**	89.7	84.3	83.7
	R@10	**95.0**	**95.0**	94.3	94.3	**98.3**	98.0	98.0	97.7	90.3	**92.0**	88.7	91.0	**93.3**	92.0	89.7	87.7
	R@20	97.0	96.7	96.3	96.3	98.7	**99.0**	**99.0**	**99.3**	93.0	**94.0**	**94.0**	93.7	**96.3**	95.3	91.7	90.7
	MRR@20	87.8	**88.2**	87.2	87.2	**93.9**	93.0	92.2	92.1	78.3	**80.0**	73.6	76.9	83.6	**83.6**	77.4	75.5
	MAP@100	87.9	**88.3**	87.2	87.3	**94.0**	93.1	92.2	92.1	78.5	**80.1**	73.7	77.0	83.7	**83.7**	77.5	75.7
	nDCG@20	90.0	**90.2**	89.3	89.4	**95.1**	94.4	93.8	93.8	81.7	**83.3**	78.4	80.9	**86.6**	86.4	80.8	79.1

Table 16: Paragraph structuring ablation. Def.=Default; Sep.=Paragraph separation; Cue=Paragraph-initial cue (enlarged first word); S+C=paragraph separation with cue. All metrics are reported as $\times 100$ (higher is better).

F.7 Context-aware Ablation Results

Dataset	Metric	CLIP-B				CLIP-L				SigLIP-B				SigLIP-L			
		Def.	**BN**	**BV**	**BT**	**Def.**	**BN**	**BV**	**BT**	**Def.**	**BN**	**BV**	**BT**	**Def.**	**BN**	**BV**	**BT**
MultiReQA	R@1	39.1	39.1	32.9	37.3	48.5	**50.3**	49.1	48.5	19.3	**28.6**	24.8	28.0	27.3	32.3	31.1	**34.8**
	R@5	**88.8**	83.2	78.3	77.6	98.1	**98.8**	97.5	98.1	60.3	60.3	54.7	**65.2**	64.6	71.4	65.8	71.4
	R@10	**95.7**	94.4	87.6	88.2	100.0	100.0	100.0	100.0	72.7	72.7	62.1	**77.6**	75.8	79.5	79.5	**80.8**
	R@20	96.3	**99.4**	95.0	96.3	100.0	100.0	100.0	100.0	74.5	80.1	80.8	**84.5**	86.3	85.1	**90.1**	85.1
	MRR@20	**59.4**	58.0	51.3	55.1	68.6	**69.6**	69.2	68.4	36.2	43.0	39.1	**44.4**	43.6	48.5	46.6	**50.6**
	MAP@100	**59.5**	58.0	51.5	55.2	68.6	**69.6**	69.2	68.4	36.9	43.5	39.6	**44.8**	43.9	49.0	46.8	**51.1**
	nDCG@20	**68.4**	67.8	61.6	64.8	76.5	**77.2**	77.0	76.3	45.4	51.7	48.5	**53.8**	53.5	57.1	56.5	**58.8**
NarrativeQA	R@1	49.5	49.4	45.6	**51.6**	71.1	70.7	70.6	**71.9**	51.1	**56.6**	51.4	56.5	**61.9**	59.6	59.5	59.1
	R@5	68.2	68.2	65.0	**69.3**	84.8	84.6	84.4	**84.9**	69.1	**72.5**	68.3	72.1	**76.7**	75.1	74.3	74.2
	R@10	74.5	74.4	71.7	**75.5**	**89.0**	88.7	88.5	88.8	75.1	**77.6**	74.1	77.3	**81.5**	80.3	79.0	79.1
	R@20	80.4	80.4	78.2	**80.8**	**92.3**	92.0	91.7	92.0	80.7	**82.6**	79.7	82.0	**85.7**	84.7	83.5	83.9
	MRR@20	57.9	57.8	54.3	**59.5**	77.3	77.0	76.9	**77.8**	59.2	**63.8**	59.0	63.6	**68.6**	66.7	66.3	65.9
	MAP@100	58.2	58.1	54.6	**59.8**	77.5	77.1	77.0	**77.9**	59.5	**64.0**	59.3	63.8	**68.8**	66.9	66.5	66.2
	nDCG@20	63.1	63.0	59.8	**64.4**	80.8	80.5	80.3	**81.1**	64.2	**68.1**	63.8	67.8	**72.5**	70.8	70.2	70.1
GoogleNQ	R@1	83.0	83.0	81.0	**83.3**	**91.3**	90.3	89.7	90.7	71.7	75.3	73.7	**76.0**	78.3	79.3	78.3	**80.0**
	R@5	93.3	92.7	91.7	**93.7**	96.7	96.0	**97.0**	96.0	85.3	86.7	83.3	**87.0**	90.3	90.3	89.3	**90.7**
	R@10	95.0	95.7	94.3	**95.7**	98.3	98.3	98.0	**98.7**	90.3	89.0	86.7	**92.0**	93.3	92.7	**93.7**	92.3
	R@20	**97.0**	96.7	96.0	96.3	98.7	**99.3**	98.7	99.0	93.0	92.3	91.7	**94.3**	96.3	94.3	94.3	**94.7**
	MRR@20	**87.8**	87.3	85.9	87.8	**93.9**	93.1	92.8	93.2	78.3	79.8	78.7	**80.9**	83.6	84.3	83.4	**85.0**
	MAP@100	**87.9**	87.4	86.0	87.9	**94.0**	93.1	92.8	93.2	78.5	79.9	78.8	**81.1**	83.7	84.3	83.4	**85.1**
	nDCG@20	**90.0**	89.5	88.3	89.8	**95.1**	94.6	94.2	94.5	81.7	82.7	81.6	**84.1**	86.6	86.7	86.0	**87.3**

Table 17: Context-aware highlighting results. Def.=Default; BN=Bold Nouns; BV=Bold Verbs; BT=Bold Topic Words. All metrics are reported as $\times 100$ (higher is better).

F.8 Noise Ablation Results

Dataset	Metric	CLIP-B				CLIP-L				SigLIP-B				SigLIP-L			
		None	Gau.	Poi.	Uni.	None	Gau.	Poi.	Uni.	None	Gau.	Poi.	Uni.	None	Gau.	Poi.	Uni.
MultiReQA	R@1	39.1	38.5	39.1	**39.8**	48.5	49.7	**51.6**	48.5	19.3	**26.7**	20.5	25.5	27.3	**28.0**	23.0	26.1
	R@5	**88.8**	84.5	87.0	88.2	98.1	**98.8**	**98.8**	98.8	60.3	61.5	50.9	**62.7**	64.6	**67.7**	67.1	67.1
	R@10	**95.7**	95.0	93.2	94.4	100.0	100.0	100.0	100.0	**72.7**	67.7	59.0	72.1	75.8	**77.0**	80.1	74.5
	R@20	96.3	**96.9**	95.0	**96.9**	100.0	100.0	100.0	100.0	74.5	77.0	70.8	**77.6**	**86.3**	85.1	88.2	83.9
	MRR@20	**59.4**	58.2	58.1	59.1	68.6	69.2	**70.2**	68.9	36.2	**41.6**	33.4	41.4	43.6	**44.5**	41.6	43.0
	MAP@100	**59.5**	58.3	58.3	59.2	68.6	69.2	**70.2**	68.9	36.9	**42.1**	34.0	42.0	43.9	**44.9**	41.9	43.4
	nDCG@20	**68.4**	67.6	67.1	68.3	76.5	76.9	**77.7**	76.7	45.4	49.8	41.9	**50.0**	53.5	**54.0**	52.5	52.6
NarrativeQA	R@1	**49.5**	48.8	47.2	49.0	71.1	71.4	70.9	**71.5**	6.1	6.0	5.8	**6.1**	51.1	52.4	37.2	**53.2**
	R@5	**68.2**	67.4	66.2	67.6	84.8	85.1	85.0	**85.1**	16.3	**16.4**	15.9	16.2	69.1	70.0	58.0	**70.9**
	R@10	**74.5**	73.8	72.7	74.1	89.0	89.3	89.0	**89.2**	**23.3**	23.2	22.5	23.2	75.1	75.9	65.7	**77.0**
	R@20	**80.4**	79.9	79.0	80.1	92.3	92.5	92.4	**92.6**	32.0	**32.1**	31.3	32.0	80.7	81.3	72.9	**82.1**
	MRR@20	**57.9**	57.2	55.9	57.4	77.3	77.6	77.2	**77.7**	**11.3**	11.2	10.8	11.2	59.2	60.4	46.6	**61.2**
	MAP@100	**58.2**	57.5	56.2	57.7	77.5	77.8	77.3	**77.8**	**12.0**	11.9	11.5	11.9	59.5	60.6	47.0	**61.5**
	nDCG@20	**63.1**	62.4	61.2	62.6	80.8	81.1	80.7	**81.2**	**15.8**	15.8	15.3	15.8	64.2	65.2	52.6	**66.1**
GoogleNQ	R@1	83.0	83.0	**83.3**	83.0	91.3	91.0	90.7	**91.7**	**20.3**	17.7	17.7	17.3	71.7	72.0	63.3	**73.0**
	R@5	**93.3**	92.0	92.7	92.0	96.7	97.0	97.0	**97.3**	**41.7**	40.0	39.7	39.7	**85.3**	84.3	79.0	85.0
	R@10	95.0	95.0	94.7	95.0	98.3	**99.0**	98.7	98.3	**55.7**	52.3	51.3	51.7	**90.3**	88.3	83.3	89.7
	R@20	**97.0**	96.3	96.3	96.7	98.7	99.0	99.0	99.0	**69.0**	67.0	65.3	67.7	93.0	93.3	85.7	**94.0**
	MRR@20	**87.8**	87.3	87.5	87.3	93.9	93.6	93.4	**94.0**	**30.8**	28.1	28.1	28.0	78.3	77.9	69.9	**78.6**
	MAP@100	**87.9**	87.4	87.6	87.4	94.0	93.6	93.4	**94.0**	**31.4**	28.8	28.8	28.6	78.5	78.1	70.3	**78.7**
	nDCG@20	**90.0**	89.4	89.6	89.5	95.1	94.9	94.7	**95.2**	**39.3**	36.7	36.4	36.8	81.7	81.4	73.6	**82.1**

Table 18: Noise ablation results. None=No noise; Gau.=Gaussian noise; Poi.=Poisson noise; Uni.=Uniform noise. All metrics are reported as $\times 100$ (higher is better).

F.9 Color Configuration Ablation Results

Model	Dataset	High Contrast			Mid Contrast			Low Contrast		
		White/Black	Black/White	Yellow/Blue	Yellow/Brn.	Blue	Grn./Pink	Pink	Cyan	Org./Purp.
CLIP-B	MultiReQA	0.6842	0.6766	0.6872	0.6846	0.6851	0.6362	0.6591	0.6706	0.6789
	NarrativeQA	0.6081	0.6305	0.6326	0.6256	0.6324	0.6167	0.6087	0.6110	0.6223
	Google NQ	0.8998	0.9083	0.8934	0.9030	0.8954	0.8960	0.8988	0.9034	0.9006
CLIP-L	MultiReQA	0.7645	0.7580	0.7638	0.7738	0.7593	0.7662	0.7553	0.7648	0.7644
	NarrativeQA	0.8109	0.8080	0.8110	0.8121	0.8072	0.8060	0.8016	0.8013	0.8066
	Google NQ	0.9506	0.9531	0.9495	0.9495	0.9511	0.9509	0.9458	0.9488	0.9477
SigLIP-B	MultiReQA	0.4537	0.4458	0.4938	0.4631	0.5244	0.5261	0.4257	0.4789	0.4563
	NarrativeQA	0.6346	0.6415	0.6626	0.6612	0.6720	0.6954	0.6809	0.6745	0.6612
	Google NQ	0.8174	0.8203	0.8379	0.8289	0.8513	0.8747	0.7881	0.8396	0.8151
SigLIP-L	MultiReQA	0.5347	0.4666	0.5429	0.5099	0.5502	0.5269	0.5263	0.5631	0.5376
	NarrativeQA	0.6543	0.7253	0.7231	0.6909	0.7101	0.7273	0.7163	0.7216	0.7120
	Google NQ	0.8656	0.8206	0.8546	0.8310	0.8543	0.8478	0.8373	0.8816	0.8562

Table 19: nDCG@20 across color configurations. Configuration abbreviations are defined in the text. Columns are ordered from highest to lowest text–background contrast (left to right). Cell shading is normalized per model (row-wise); darker indicates higher nDCG@20 within the same model.

G Detailed Paraphrasing Results

IR Dataset	Transform	IR (nDCG@20)				QA Task	QA (×100)			
		MiniLM	bge-L	gte-L	Rob		*LLMs*		*VLMs*	
							4o-mini	Llama-8b	Qwen3-VL	Glyph
Google NQ	Original	96.9	96.8	97.6	10.9	Memory	32.1	39.3	45.0	47.0
	Lexical	97.2	96.7	97.8	11.4		30.7	36.4	40.6	42.4
	Syntactic (active)	95.3	96.7	97.8	10.1		7.5	8.2	9.9	10.3
	Syntactic (passive)	96.9	96.9	97.3	12.2		28.5	32.5	37.7	39.4
	Style (compression)	97.5	97.3	97.9	6.9		10.2	9.9	12.3	14.9
	Style (expansion)	96.1	97.4	97.8	9.2		7.3	7.5	8.7	9.4
	Style (formal)	96.4	97.0	98.0	8.5		7.4	8.8	10.6	11.1
	Style (entertainment)	95.0	97.5	98.2	11.3		8.4	8.9	9.1	9.5
	Highlight (nouns)	95.0	97.1	97.5	7.3		34.4	39.4	49.1	46.4
	Highlight (verbs)	96.9	97.7	97.8	9.3		33.2	39.8	46.2	48.9
	Highlight (topic)	96.6	97.1	97.7	12.7		33.1	38.7	45.2	45.2
MultiReQA	Original	69.9	77.8	77.2	6.3	Reasoning	41.0	34.5	56.0	56.5
	Lexical	71.5	77.5	77.0	9.4		40.0	38.0	55.5	55.0
	Syntactic (active)	71.5	76.5	77.8	8.3		38.5	37.0	51.0	51.5
	Syntactic (passive)	71.2	77.2	77.6	7.0		44.5	43.0	59.0	59.5
	Style (compression)	71.5	77.5	77.8	9.6		13.5	13.5	19.0	16.0
	Style (expansion)	70.2	76.9	77.3	9.7		25.0	28.0	36.5	34.5
	Style (formal)	71.5	75.7	77.3	9.1		37.0	40.5	50.5	51.0
	Style (entertainment)	72.8	78.3	77.4	9.7		33.5	32.5	43.0	43.4
	Highlight (nouns)	67.4	73.7	75.4	5.9		31.5	28.5	54.0	54.5
	Highlight (verbs)	71.1	77.7	77.3	7.1		41.5	32.0	60.5	58.0
	Highlight (topic)	69.2	77.2	76.8	6.2		40.5	31.5	58.5	57.0
NarrativeQA	Original	82.6	79.3	78.4	2.8	Retrieval	98.5	99.5	100.0	95.5
	Lexical	82.3	79.1	78.2	2.9		97.3	97.8	99.8	95.5
	Syntactic (active)	82.8	79.4	78.0	3.5		45.3	45.3	45.3	45.3
	Syntactic (passive)	81.9	78.8	78.0	3.0		99.0	100.0	100.0	99.5
	Style (compression)	81.3	77.1	78.1	3.3		6.8	6.8	6.8	6.8
	Style (expansion)	81.0	76.8	74.8	3.1		7.5	6.8	7.5	8.8
	Style (formal)	82.7	78.7	78.0	3.5		6.8	4.3	6.5	6.5
	Style (entertainment)	78.3	72.3	71.7	3.5		12.0	11.5	11.5	12.0
	Highlight (nouns)	78.5	73.7	77.3	1.7		99.0	99.8	99.8	98.0
	Highlight (verbs)	81.4	73.1	77.7	2.2		98.5	99.3	99.8	97.5
	Highlight (topic)	81.2	74.9	77.1	2.8		99.5	99.5	100.0	97.0

Table 20: Paraphrasing results on information retrieval (nDCG@20, left) and question answering (right). IR evaluates text encoders across three retrieval datasets (Rob stands for RoBERTa). QA evaluates LLMs (GPT-4o-mini, Llama-8B) and VLMs (Qwen3-VL, Glyph) on Memory (ROUGE-L), Reasoning (contains_all), and Retrieval (contains_all). * Values are estimated using linear scaling from the Qwen3-VL baseline trend.

H Artifacts, Licensing, and Reproducibility

All scientific artifacts used or created in this work are handled strictly in accordance with their original licensing terms and intended research purposes.

External Datasets and Models. Our evaluation relies entirely on publicly available benchmarks and open-source models. The information retrieval and question answering datasets (MultiReQA, NarrativeQA, and Google NQ) along with the baseline models (CLIP, Qwen-VL, InternVL, and bge-L) are utilized under their respective open-source agreements (such as Apache 2.0, MIT, or custom research licenses).

Intended Use Compliance. We explicitly verify that our utilization of these datasets and baseline models is fully consistent with their intended use as academic evaluation benchmarks and public research foundations. Furthermore, the VTAOpt framework introduced in this work is created and distributed solely for scientific research in vision-language processing and document optimization. Its intended use remains strictly compatible with the original access conditions of the underlying artifacts, and we discourage its deployment outside of fair academic or authorized research contexts.

Code Availability and Licensing. To support reproducibility and foster future research into Visual Text Augmentation, the full implementation details, experimental configurations, and the core VTAOpt Bayesian optimization framework will be made publicly available on GitHub under the MIT License upon publication.

I Computing Infrastructure and Budget

To ensure complete experimental transparency and reproducibility, we report our hardware infrastructure and cumulative computational budget. All optimization searches and evaluations were conducted using a localized compute cluster equipped with NVIDIA A100 (80GB) GPUs. The total computational budget required to execute the Bayesian optimization loops across our benchmarks for all VLMs amounted to approximately 60 total GPU hours. Because VTAOpt runs exclusively at inference time without requiring any gradient steps or weights tuning, its active carbon footprint and compute requirements are significantly lower than traditional model fine-tuning methods.

J The Use of Large Language Models

In this work, we used LLMs as supportive tools rather than research contributors. Specifically, LLMs assisted with polishing the writing for clarity and readability and with refining prompt designs for our experiments. However, they were not involved in research ideation, conceptual framing, or substantive writing of the paper. All outputs produced with LLM assistance were carefully reviewed and verified by the authors, who take full responsibility for the content of this research.