Does Text Have to Be Tokenized?

Pixel-Based Language Understanding and Token Efficiency in Multimodal LLMs

26.08.06 (목) / 홍성태

ghdchlwls123@korea.ac.kr

PixelWorld: How Far Are We from Perceiving Everything as Pixels?*

Zhiheng Lyu[1,2] Xueguang Ma[1] Wenhu Chen[1,2]
[1]University of Waterloo [2]Vector Institute, Toronto
{z63lyu,x93ma,wenhuchen}@uwaterloo.ca

TMLR 2025 Oct

Why Separate Pixels and Tokens?

Representation mismatch in multimodal agents

현재 VLM은 image를 pixel patch로, text를 discrete token으로 처리하는 이중 파이프라인

시각 정보와 언어 정보가 한 화면에 결합될수록 전처리·정렬·레이아웃 보존의 복잡성 증가

Separate Processing

- image encoder와 tokenizer의 분리된 표현 공간
- OCR 결과와 원본 layout 사이의 불일치
- modality-specific preprocessing의 누적 비용

Agentic Environments

- 웹페이지·슬라이드·문서 내부의 긴밀한 시각–언어 결합
- raw camera pixel에서 바로 행동해야 하는 실제 환경
- 텍스트 순서만으로 복원하기 어려운 공간적 관계

Research Question

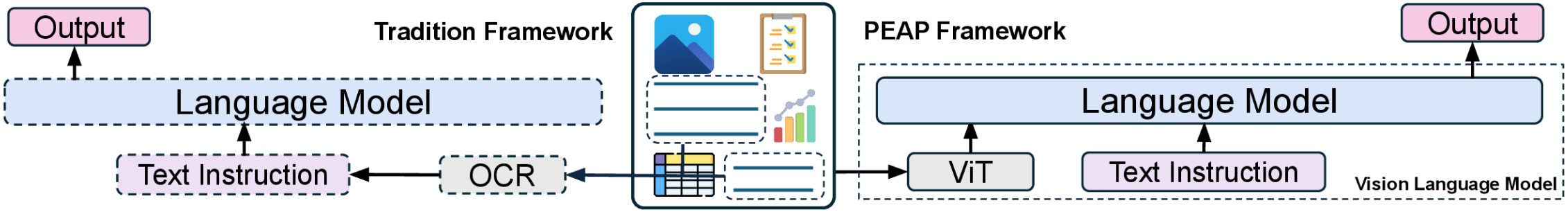
- 모든 입력을 하나의 pixel space로 통일할 가능성
- 명시적 tokenization 없이 유지되는 semantic capability
- 통합 표현에서 발생하는 reasoning·efficiency 한계

현재 VLM은 이미지와 텍스트를 서로 다른 표현으로 처리하기 때문에, 한 화면에서 두 정보가 긴밀하게 연결될수록 OCR 오류와 정렬 비용이 증가.

➔ 텍스트까지 픽셀로 통일했을 때 공간 정보와 언어 이해를 함께 보존할 수 있는가에 대한 검증.

Perceive Everything as Pixels

A single pixel space



Unified Input

text · table · code · diagram
→ rendered pixels

Vision Encoder

ViT patch encoding
without explicit OCR

Language Decoder

instruction following
and answer generation

PEAP는 텍스트·표·코드·이미지를 하나의 화면으로 렌더링한 뒤 ViT와 언어 디코더만으로 처리하는 구조.

➔ 새로운 모델을 학습하기보다, 현재 VLM이 명시적 OCR과 tokenization 없이 언어 의미를 어디까지 복원하는지 진단

Benchmark & Data construction

Text-only (6)

- GLUE · SuperGLUE · ARC의 의미 이해
- MMLU-Pro · GSM8K의 지식·수학 추론
- MBPP의 programming capability

Structured (1)

- TableBench의 fact checking과 data analysis
- numerical reasoning과 visualization
- text · image · semi 입력 형식 비교

Multimodal (4)

- MathVerse · MMMU-Pro의 시각 추론
- SlidesVQA의 slide understanding
- Wiki-SS의 long-context document QA

Original Input
text prompt
table / document

→

Modality Transfer
image synthesis
or OCR extraction

→

Paired Evaluation
same instruction
two representations

Rendering Variation

- adaptive width 512–1024 px, fixed height 256 px
- font size 15–25 pt, padding 5–30 px
- 문장 길이에 따른 화면 밀도 변화

Robustness Noise

- radial·horizontal·vertical distortion
- Multi-Gaussian과 high-frequency noise
- 실제 camera capture의 변형 모사

Evaluation Protocol

- instruction sensitivity를 줄이는 zero-shot 평가
- 동일 내용의 modality transfer에 집중

Experiments

Token input versus pixel input

Model

- Qwen2-VL- 2B
- Phi-3.5-Vision-3.2B
- Qwen2VL- 7B
- Gemini-Flash
- GPT-4o

Classification

- MMLU-Pro·ARC·MathVerse의 accuracy
- GLUE·SuperGLUE의 task-specific metric

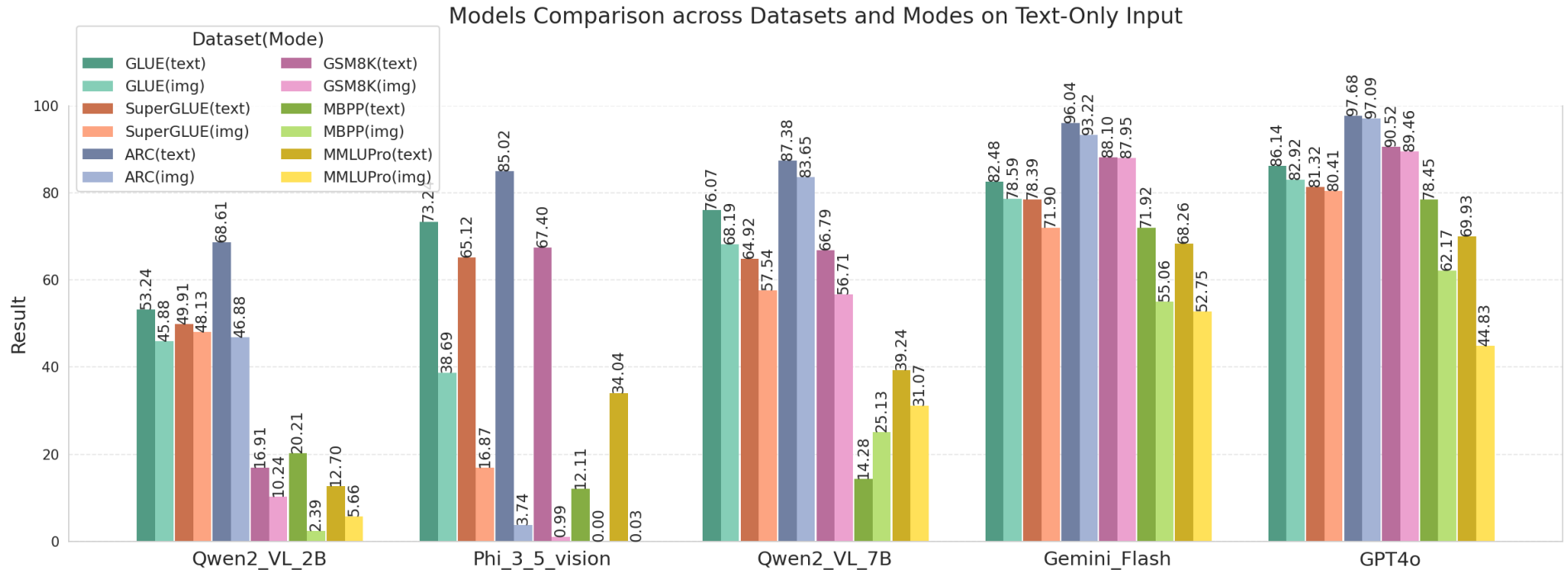
Generation

- GSM8K의 exact match accuracy
- MBPP의 pass@1 code execution rate

QA

- SlidesVQA·WikiSS·TableBench의 ROUGE-L
- 예측과 정답의 longest common subsequence

Results



Semantic transfer

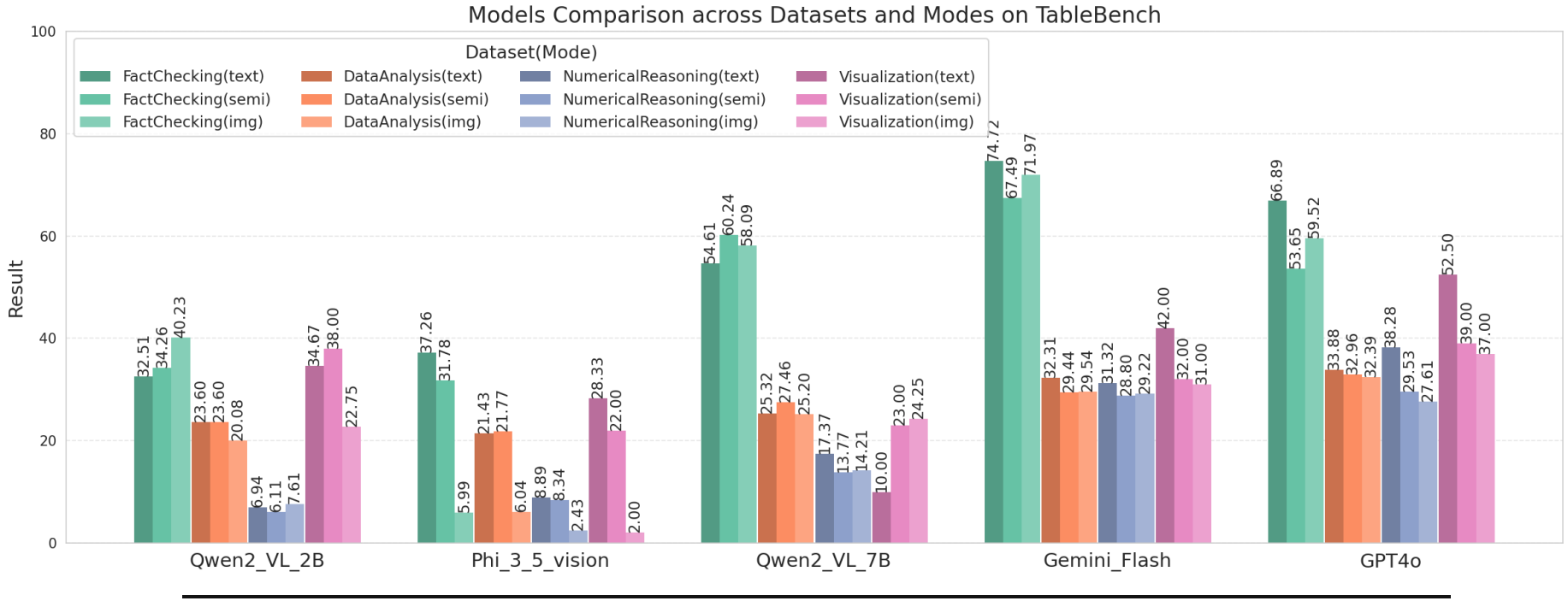
- GLUE·SuperGLUE·ARC에서 비교적 작은 modality gap
- 문장·문단의 global semantics를 ViT가 부분적으로 포착

Reasoning gap

- MMLU-Pro·GSM8K·MBPP의 수학·코드 성능 급락
- GPT-4o도 MMLU-Pro에서 25점 이상 하락

그래프를 왼쪽에서 오른쪽으로 보면 모델이 커질수록 text–image 간극이 감소하지만, task 종류에 따른 차이는 계속 유지.
즉 문장 의미는 픽셀에서도 상당 부분 복원되지만, 수학과 코드에 필요한 정확한 기호 조작은 별도의 학습과 구조가 필요한 결과.

Results



Semantic tasks

- Fact Checking·Data Analysis에서 text와 유사하거나 더 높은 image score
- 표의 행·열 배치가 semantic relation의 시각적 단서

Reasoning tasks

- Numerical Reasoning의 정확한 수치 조작에서 큰 하락
- Visualization code 생성에서 기호 정확성의 병목

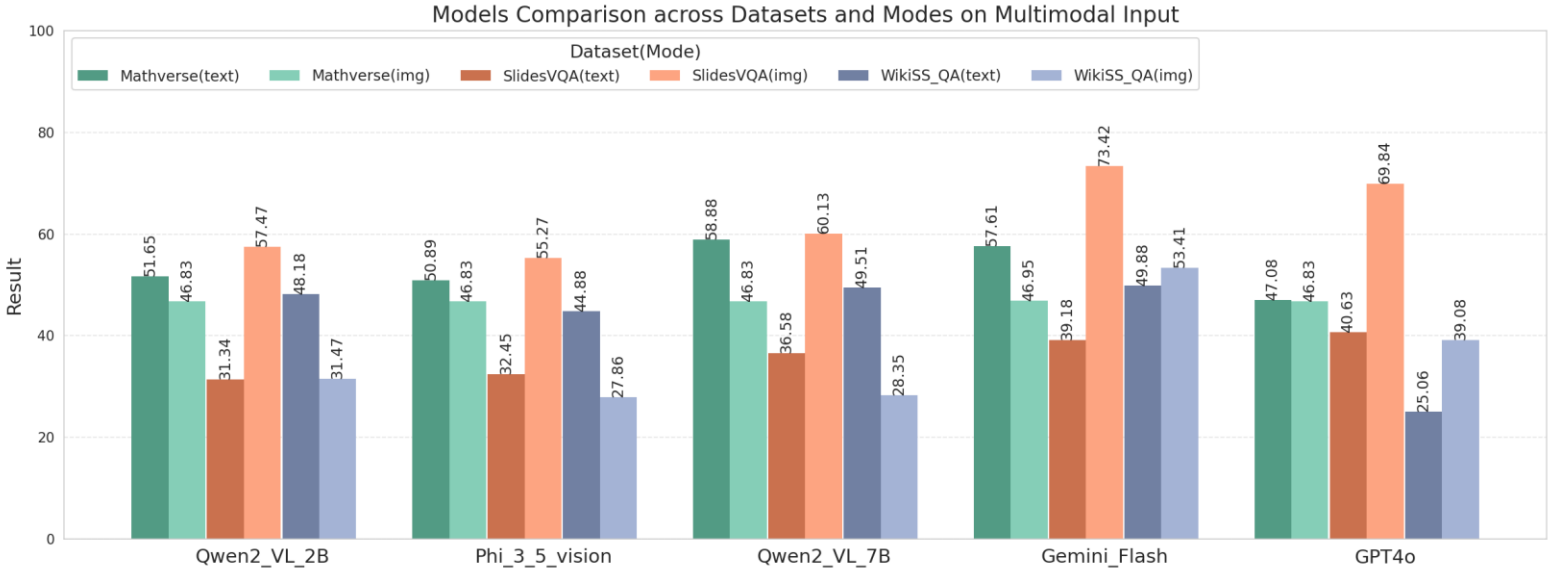
Semi setting

- text 질문 + image 표가 두 단일 표현보다 자주 낮은 성능
- 분리된 encoder 사이의 cross-modal alignment 부담

표의 행과 열을 그대로 이미지로 보존하면 fact checking과 data analysis에는 유용한 시각적 단서가 제공. 그러나 numerical reasoning과 visualization에서는 정확한 계산과 코드 생성이 필요하고, 질문과 표를 서로 다른 표현으로 주는 semi 방식에는 추가 정렬 병목까지 발생.

Results

Multimodal inputs benefit from visual context



Visual context

- SlidesVQA에서 모든 모델이 text-only baseline을 상회
- 위치·크기·도형 관계가 질문의 모호성을 감소

Complex reasoning

- MathVerse의 multi-step deduction에서 지속되는 병목
- 시각 단서가 perception을 돕지만 reasoning을 대체하지 못함

Scale effect

- SlidesVQA: Gemini +34.24점, Qwen2VL-7B +23.55점
- 큰 모델일수록 multimodal evidence 활용 증가

시각 문맥은 지각을 개선하지만 복잡한 추론 병목은 유지

SlidesVQA에서는 이미지가 위치·크기·도형 관계를 보존하여 모든 모델의 모호성을 줄이고, 큰 모델일수록 그 이점을 더 크게 활용.
반면 MathVerse의 다단계 문제에서는 시각 정보가 문제 이해를 돕더라도 논리적 추론 과정을 대신하지 못하는 한계.

Analysis

Similar attention, redundant patches

두 입력이 질문과 핵심 문장에 유사한 attention을 보인다는 사실은 언어 행동의 부분적 전이를 의미.

동시에 빈 이미지 영역에도 attention이 분산되므로, 표현이 동일하다는 결론이 아니라 불필요한 visual token을 줄일 여지가 크다는 해석.

Conclusion

Pixels work for semantics, not yet for everything

Semantic understanding

- 문장·문단 과제에서 token baseline과 유사한 성능
- 명시적 tokenization 없는 의미 이해 가능성
- 유사한 global attention을 통한 부분적 language transfer

Reasoning boundary

- 수학·논리·코드에서 지속되는 큰 성능 하락
- CoT가 완화하지만 제거하지 못하는 symbolic gap
- 모델 규모만으로 해결되지 않는 exact operation

Multimodal potential

- 슬라이드·문서의 spatial context 보존
- OCR noise와 cross-modal alignment 비용 감소 가능성

PixelWorld는 '모든 것을 픽셀로 통일할 수 있다'는 결론이 아니라,
현재 VLM의 가능 범위와 실패 지점을 측정하는지 검증

Text or Pixels? It Takes Half: On the Token Efficiency of Visual Text Inputs in Multimodal LLMs

Yanhong Li[*]
Allen Institute for AI
yanhongl@allenai.org

Zixuan Lan[*]
University of Chicago
zixuanlan@uchicago.edu

Jiawei Zhou
Stony Brook University
jiawei.zhou.1@stonybrook.edu

EMNLP 2025 Findings

Long Context Is Expensive Where It Matters

비용은 context capacity가 아닌 실제 decoder 처리 길이에서 발생

Transformer의 self-attention과 input length에 비례하여 증가하는 계산량·메모리·지연 시간

계산 비용의 병목

- 긴 문맥 처리에 따른 attention 계산량과 KV-cache 부담 증가
- 동일 요청에서도 decoder가 읽는 입력 길이에 따라 증가하는 serving cost

Context window 확장만으로 해결되지 않는 문제

- 더 큰 context window는 수용 가능 길이의 확대일 뿐 처리 비용의 제거가 아님
- 사용하지 않는 정보까지 모두 decoder token으로 전달하는 구조적 비효율

기존 접근의 한계

- retrieval·summary·token pruning에 의한 정보 선택 또는 삭제 가능성
- compression model 학습과 task-specific tuning에 대한 추가 의존성

핵심 목표

- 정보 손실 없이 decoder가 처리하는 sequence length를 줄이는 입력 압축
- 새로운 compressor가 아닌 기존 multimodal pathway의 활용 가능성 검증

동일 정보를 유지하면서 token을 얼마나 줄일 수 있는가???

Visual Inputs Already Embody a Form of Token Compression

Vision encoder를 통한 입력 표현의 축약

- 텍스트가 포함된 전체 이미지를 제한된 개수의 visual embedding으로 변환
- 동일한 정보를 원래 text token보다 짧은 embedding sequence로 전달

Decoder context length의 직접 감소

- 모든 text token 대신 k개의 visual token과 query만 decoder에 입력
- attention 및 KV-cache가 의존하는 decoder-side sequence length의 감소

새로운 압축 모델이 아닌 modality shifting

- 별도의 compressor 학습이나 추가 supervision 없이 기존 pathway 활용
- vision encoder의 projection layer가 제공하는 내재적 압축 능력의 측정

이 논문의 측정 질문

- 시각적 text density를 높여도 downstream performance를 유지
- text-token tolerance와 실제 latency trade-off에 대한 정량적 분석

Existing visual pathway = an untrained context-compression channel

Rendering Long Passages as a Single Image

두 가지 입력 파이프라인

- Text-only: context와 query를 모두 decoder에 text token으로 전달
- Hybrid: context를 하나의 이미지로 렌더링하고 query만 text로 유지

이미지 기반 압축 흐름

- Vision encoder가 전체 이미지를 고정된 visual token sequence로 변환
- 동일 정보의 보존과 decoder input length의 감소를 동시에 추구

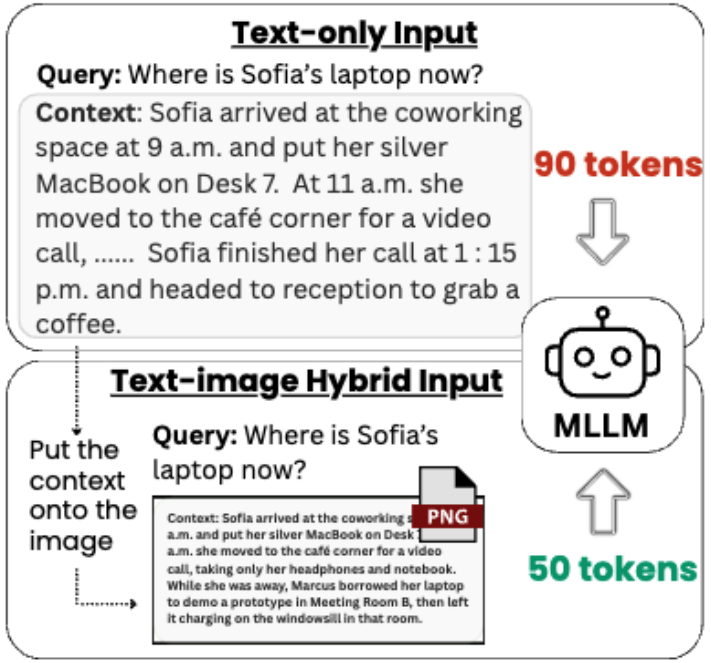


Figure 1: Illustration of our *text-as-image* compression pipeline. Instead of feeding the entire 90-token context to the model (top), we convert the context into a single image and supply only the textual query alongside the image (bottom). The multimodal LLM (MLLM) reads the context from the image, so it processes just 50 visual tokens as input to the LLM decoder—cutting token usage by nearly half while still providing all the information needed to answer the question.

From m Text Tokens to k Visual Tokens

Text-only input budget

$$T_{text} = m + |q|$$

문맥 m개와 query token을 decoder에 전달

Text-image hybrid budget

$$T_{img} = k + |q|$$

문맥을 대체한 k개 visual token과 query만 전달

Compression ratio

$$\rho = T_{text} / T_{img} \approx m / k$$

짧은 query 조건에서 text-to-visual token 비율

m : 원문 텍스트를 구성하는 context text-token count

k : vision encoder와 decoder가 전달받는 visual-token budget

m* : text-only baseline 대비 성능 저하를 허용 범위 안에 유지하는 최대 text-token count

How Much Can We Compress Without Losing Performance?

Text-only

- context m개 + query를 text token으로 직접 전달
- decoder가 원문 전체를 처리하는 baseline

Text-image hybrid

- context를 image로 렌더링하고 query만 text로 유지
- visual-token budget k와 text density를 변화시키는 조건

Models

- GPT-4.1-mini
- Qwen2.5-VL-72B-Instruct
- parameter update와 fine-tuning 부재

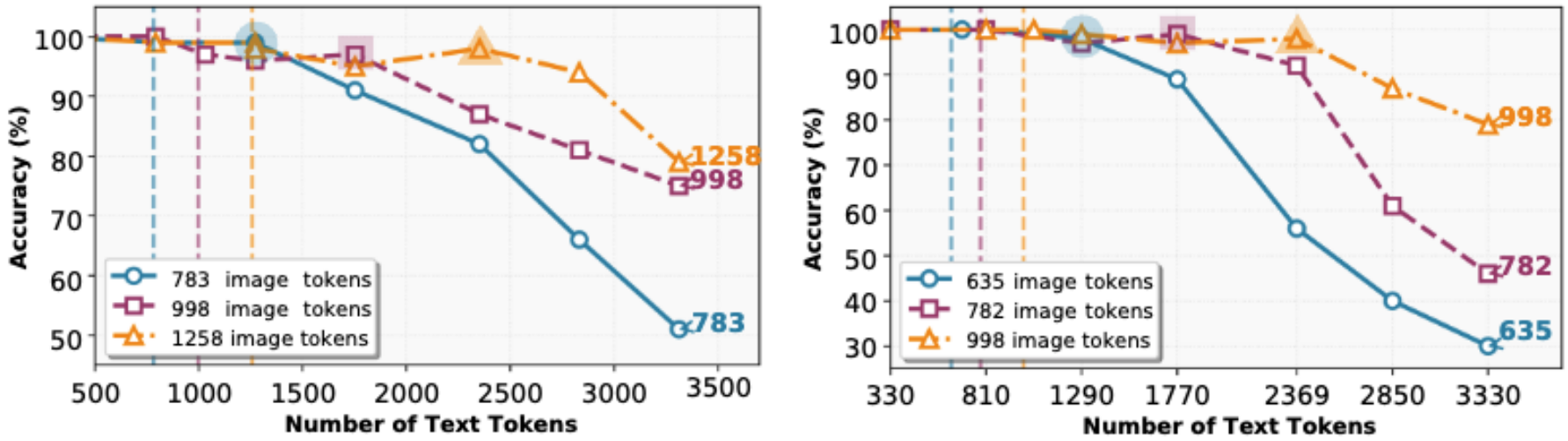
Benchmark

- RULER S-NIAH long-context retrieval
- 긴 distractor context 속 하나의 target number 추출
- 100개 random passage 기반 exact accuracy

Metrics

- task accuracy와 text-token tolerance m*
- decoder token usage와 compression ratio
- end-to-end wall-clock latency

Accuracy Holds Until a Critical Point



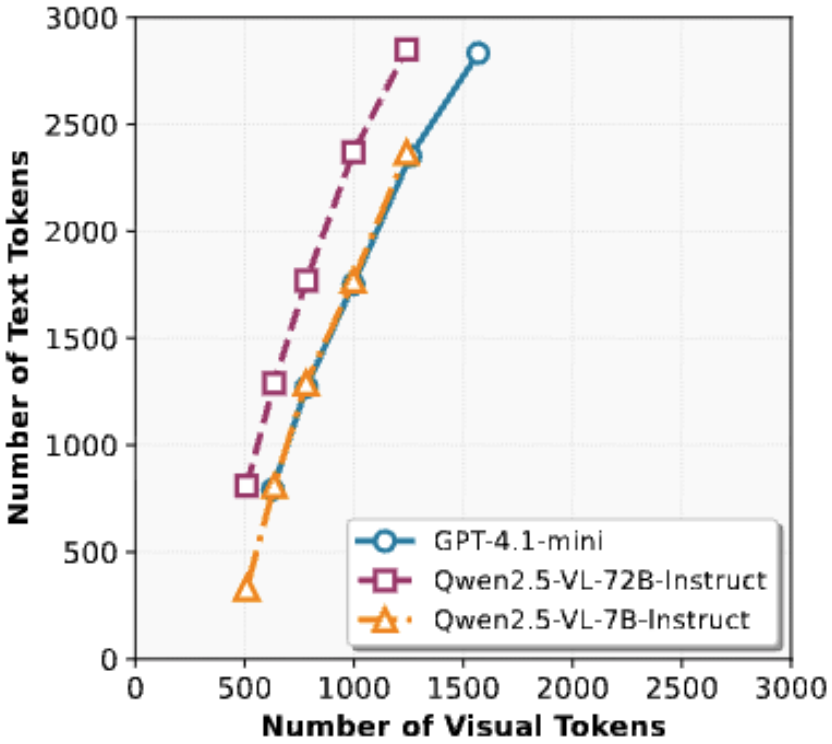
(a) GPT-4.1-mini

(b) Qwen2.5-VL-72B-instruct

고정 visual-token budget k에서 rendered text-token count m을 증가시키며 accuracy 변화 측정

- text-only baseline과 유사한 안정 구간 이후 model-specific critical point에서 급격한 성능 저하
- 더 큰 k는 더 높은 text density를 수용하지만 허용 가능한 m*는 모델별 차이

The Text-Token Tolerance



- 주어진 visual-token budget k에서 text-only baseline 대비 3 %p 이내 accuracy를 유지하는 최대 m을 m*(k)로 정의
- 모든 모델에서 m*(k)는 k에 따라 거의 선형적으로 증가
- 동일한 k에서도 vision encoder와 model architecture에 따른 tolerance 차이
- 점선 1/2 reduction guide 위에 위치하는 실험 곡선

➔ 약 2배 수준의 decoder-token compression을 시사하는 operating point

➔ 목표 성능과 budget에 맞춘 모델별 m*(k) curve 측정 필요

Nearly Half the Decoder Context, Same Accuracy

38–58%

RULER S-NIAH에서 감소한 decoder token footprint

Hybrid accuracy 97–99% · Text-only reference 100% · accuracy와 latency의 공동 해석

Model	Text–Image (hybrid)			Text-only		
	Image size	k	Acc. (%)	$m^\star(k)$	Acc. (%)	$k/m^\star(k)\downarrow$
GPT	600×800	783	99	1,272.4	100	0.61
	600×1000	998	97	1,752.5	100	0.57
	750×1000	1,258	98	2,352.2	100	0.53
QWEN	600×800	635	98	1,289.6	100	0.49
	600×1000	782	99	1,769.7	100	0.44
	750×1000	998	98	2,369.4	100	0.42

(a) **RULER accuracy and token statistics.** k is the visual token count, $m^\star(k)$ the maximum *text-token tolerance*, and $k/m^\star(k)$ the relative token footprint. Text-image input reduces the decoder tokens by 38–58%.

Model	k	Time_{img} (s)↓	$\text{Time}_{\text{text}}$ (s)↓
GPT*	783	1.29	0.60
	998	1.75	0.61
	1,258	1.95	0.58
QWEN	635	2.81	3.61
	782	3.04	4.16
	998	3.35	5.09

(b) **End-to-end latency.** Hybrid inputs add modest overhead for GPT but *reduce* total time for Qwen thanks to smaller decoder contexts. *measured from API responses; see Appendix A.

Table 1: RULER S-NIAH long-context retrieval accuracy, text-as-image compression statistics, and model latency. GPT refers to GPT-4.1-mini and QWEN denotes Qwen2.5-VL-72B-Instruct.

A Competitive—and Orthogonal—Alternative

Decoder token을 약 절반 이상 줄이면서 유지되는 summarization 품질

- GPT 67%, Qwen 62% token reduction · 유사 compression rate의 Select-Context와 LLMLingua-2 대비 높은 reported metrics
- task-specific compressor 학습 없이 확보한 competitive result와 token pruning 이후 결합 가능한 orthogonal axis

Model	Method	Remaining $k\downarrow$	BERTScore	ROUGE-L	ROUGE-1	ROUGE-2	ROUGE-L_{sum}
GPT	Baseline (text-only)	m=693	**86.25**	**16.26**	**23.78**	**8.60**	**18.91**
	Text-as-image (ours)	225 (−67%)	85.33	15.31	21.98	7.40	17.75
	Select-Context	295 (−57%)	85.01	12.79	18.82	5.30	15.71
	LLMLingua-2	265 (−62%)	85.25	13.75	20.42	7.00	16.60
QWEN	Baseline (text-only)	m=726	**86.37**	**17.70**	**25.18**	**9.47**	**20.77**
	Text-as-image (ours)	279 (−62%)	85.64	15.53	23.28	7.54	19.16
	Select-Context	181 (−75%)	84.60	13.40	20.36	5.13	17.63
	LLMLingua-2	258 (−64%)	85.46	15.32	22.95	7.09	18.94

Table 2: CNN/DailyMail document summarization with token compression baselines. GPT refers to GPT-4.1-mini and QWEN denotes Qwen2.5-VL-72B-Instruct. m is the uncompressed text length, k the remaining decoder tokens after compression. Percentage shows token reduction relative to m. At *similar* compression rates, rendering the document as an image beats both token-level baselines on every metric.

해석: 유사 압축률에서 모든 보고 지표 개선

한계: task-specific SOTA 주장보다 보완적 축

Conclusion

핵심 문제의식

- 긴 text context가 multimodal LLM의 decoder token을 과도하게 소모
- vision encoder를 통한 동일 정보의 더 짧은 decoder representation 가능성

Text-as-image의 설계

- context를 하나의 이미지로 렌더링하고 query만 text로 유지하는 hybrid input
- m, k, m*, ρ로 측정하는 model-specific quality-efficiency operating point

실험 결과

- RULER에서 accuracy 유지와 38–58% decoder-token reduction
- 경쟁력 있는 summarization 품질과 약 62–67% token reduction

한계와 조건

- m* 초과 시 급격한 성능 저하와 고정 압축률의 부재
- 모델·해상도·텍스트 밀도·배포 환경별 실제 측정의 필요성

Thank You

[qasper]

		Qwen3.6-35B-A3B		Gemma-4-26B-A4B	
		score	reduction_pct	score	reduction_pct
	text	51.45	5206 (0.00%)	47.73	5181 (0.00%)
font10	1.0	41.32	2094 (61.96%)	45.9	3248 (38.69%)
	1.1	43.18	2375 (56.36%)	45.72	3617 (31.30%)
	1.2	45.8	2486 (54.15%)	44.58	3749 (28.65%)
font11	1.0	45.26	2711 (49.67%)	47.48	4050 (22.64%)
	1.1	46.46	3030 (43.33%)	48.32	4544 (12.75%)
	1.2	47.41	3202 (39.89%)	45.76	4763 (8.38%)
font12	1.0	48.72	3013 (43.67%)	48.07	4522 (13.19%)
	1.1	45.35	3407 (35.82%)	47.93	5031 (3.01%)
	1.2	45.33	3741 (29.17%)	46.5	5458 (-5.53%)

[narrativeqa]

		Qwen3.6-35B-A3B		Gemma-4-26B-A4B	
		score	reduction_pct	score	reduction_pct
	text	32.14	30704 (0.00%)	30.98	30746 (0.00%)
font10	1.0	20.93	8153 (73.70%)	27.56	11617 (62.44%)
	1.1	26.09	9381 (69.69%)	27.43	13385 (56.66%)
	1.2	25.9	9858 (68.13%)	28.87	14087 (54.37%)
font11	1.0	27.61	10811 (65.02%)	26.94	15260 (50.55%)
	1.1	29.42	12232 (60.37%)	28.39	17314 (43.84%)
	1.2	28.36	12948 (58.03%)	27.52	18405 (40.28%)
font12	1.0	30.5	12137 (60.69%)	28.58	17251 (44.05%)
	1.1	27.96	13833 (55.14%)	28.75	19482 (36.76%)
	1.2	27.96	15406 (50.00%)	27.98	21882 (28.93%)

[multi_news]

		Qwen3.6-35B-A3B		Gemma-4-26B-A4B	
		score	reduction_pct	score	reduction_pct
	text	23.41	2699 (0.00%)	22.57	2799 (0.00%)
font10	1.0	22.29	1097 (60.41%)	22.24	1894 (32.93%)
	1.1	22.8	1249 (54.67%)	22.02	2097 (25.54%)
	1.2	22.74	1300 (52.74%)	22.21	2151 (23.57%)
font11	1.0	22.59	1425 (48.04%)	22.4	2325 (17.25%)
	1.1	23.12	1582 (42.14%)	22.24	2623 (6.41%)
	1.2	23.19	1684 (38.29%)	22.14	2742 (2.06%)
font12	1.0	22.98	1568 (42.66%)	22.2	2590 (7.59%)
	1.1	23.35	1772 (34.96%)	21.9	2815 (-0.60%)
	1.2	23.41	1959 (27.92%)	22.24	3114 (-11.48%)

[multifieldqa_en]

		Qwen3.6-35B-A3B		Gemma-4-26B-A4B	
		score	reduction_pct	score	reduction_pct
	text	58.9	7234 (0.00%)	54.7	7105 (0.00%)
font10	1.0	48.23	2360 (68.00%)	56.83	3693 (48.47%)
	1.1	48.18	2709 (63.12%)	54.76	4050 (43.39%)
	1.2	50.27	2835 (61.36%)	54	4288 (40.02%)
font11	1.0	51.96	3104 (57.62%)	54.15	4610 (35.44%)
	1.1	55.83	3485 (52.30%)	53.93	5239 (26.50%)
	1.2	55.24	3710 (49.16%)	55.58	5532 (22.34%)
font12	1.0	51.65	3446 (52.85%)	55.75	5181 (27.33%)
	1.1	53.25	3937 (46.00%)	54.48	5775 (18.89%)
	1.2	55.62	4359 (40.11%)	55.3	6434 (9.52%)

[triviaqa]

		Qwen3.6-35B-A3B		Gemma-4-26B-A4B	
		score	reduction_pct	score	reduction_pct
	text	92.18	12362 (0.00%)	87.96	12185 (0.00%)
font10	1.0	91.47	4443 (68.43%)	85.18	6236 (52.17%)
	1.1	91.47	4990 (63.71%)	85.45	7031 (45.20%)
	1.2	91.57	5206 (61.84%)	85.95	7332 (42.56%)
font11	1.0	91.47	5638 (58.10%)	86.06	7890 (37.67%)
	1.1	91.47	6270 (52.65%)	84.89	8746 (30.16%)
	1.2	91.07	6603 (49.77%)	85.56	9258 (25.67%)
font12	1.0	91.47	6235 (52.95%)	86.58	8712 (30.45%)
	1.1	91.22	6994 (46.39%)	85.35	9755 (21.31%)
	1.2	92.07	7695 (40.34%)	84.67	10746 (12.62%)

[samsum]

		Qwen3.6-35B-A3B		Gemma-4-26B-A4B	
		score	reduction_pct	score	reduction_pct
	text	41.31	9507 (0.00%)	41.43	10027 (0.00%)
font10	1.0	35.58	2768 (72.28%)	38.47	4137 (59.90%)
	1.1	35.62	3147 (68.23%)	38.93	4665 (54.52%)
	1.2	35.52	3305 (66.53%)	38.63	4921 (51.92%)
font11	1.0	35.82	3607 (63.28%)	38.8	5311 (47.96%)
	1.1	36.31	4050 (58.54%)	39.39	5952 (41.44%)
	1.2	36	4277 (56.11%)	38.88	6257 (38.34%)
font12	1.0	35.93	4011 (58.95%)	39.79	5909 (41.87%)
	1.1	36.48	4550 (53.18%)	39.71	6608 (34.76%)
	1.2	36.53	5025 (48.07%)	40.14	7290 (27.83%)

[qmsum]

		Qwen3.6-35B-A3B		Gemma-4-26B-A4B	
		score	reduction_pct	score	reduction_pct
	text	22.55	14325 (0.00%)	22.91	14309 (0.00%)
font10	1.0	22.01	4440 (69.38%)	21.67	6413 (55.48%)
	1.1	22.16	5098 (64.76%)	21.89	7371 (48.75%)
	1.2	22.33	5360 (62.92%)	21.65	7731 (46.22%)
font11	1.0	22.67	5888 (59.22%)	22.85	8531 (40.60%)
	1.1	22.66	6629 (54.02%)	22.14	9520 (33.65%)
	1.2	22.43	7037 (51.16%)	22.58	10172 (29.07%)
font12	1.0	22.19	6580 (54.36%)	22.37	9487 (33.89%)
	1.1	22.88	7510 (47.83%)	22.8	10762 (24.93%)
	1.2	23.1	8357 (41.89%)	22.93	11985 (16.33%)

[gov_report]

		Qwen3.6-35B-A3B		Gemma-4-26B-A4B	
		score	reduction_pct	score	reduction_pct
	text	31.1	10628 (0.00%)	28.78	10603 (0.00%)
font10	1.0	30.55	4246 (60.33%)	28.27	6275 (41.02%)
	1.1	30.75	4872 (54.41%)	28.46	7142 (32.80%)
	1.2	30.42	5128 (51.98%)	28.27	7501 (29.39%)
font11	1.0	30.53	5620 (47.34%)	28.53	8169 (23.07%)
	1.1	30.55	6352 (40.41%)	28.69	9177 (13.51%)
	1.2	30.85	6727 (36.87%)	28.68	9748 (8.10%)
font12	1.0	30.82	6311 (40.80%)	28.67	9124 (14.02%)
	1.1	30.66	7185 (32.55%)	28.75	10318 (2.70%)
	1.2	30.69	7992 (24.91%)	28.6	11453 (-8.05%)