『2025 여름 세미나』

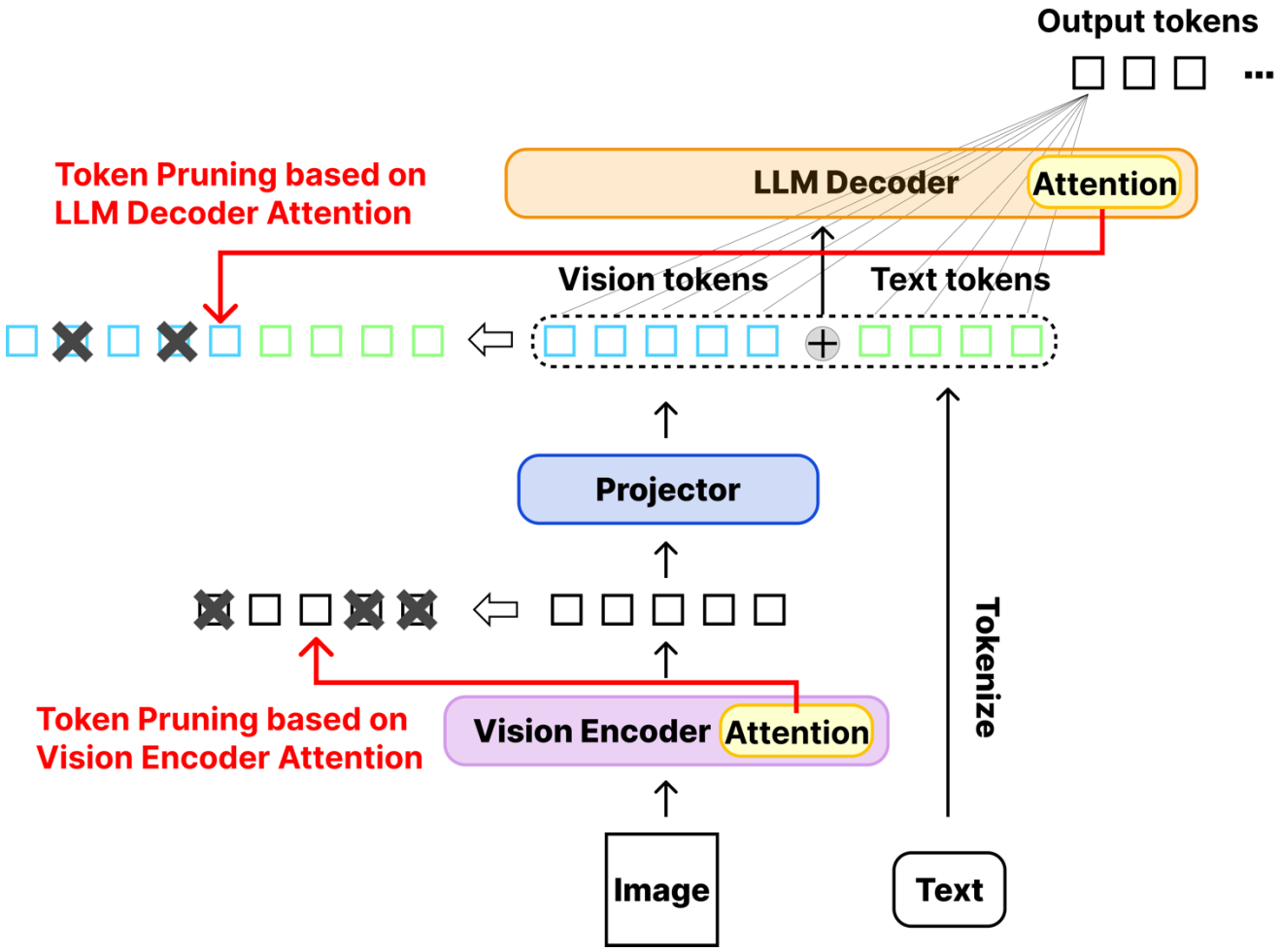
Vision Token Reduction in LVLMs

Pusan National University
Dept. of Electronics Engineering
Computer Vision & Signal Processing Lab.

Presented By
김소현

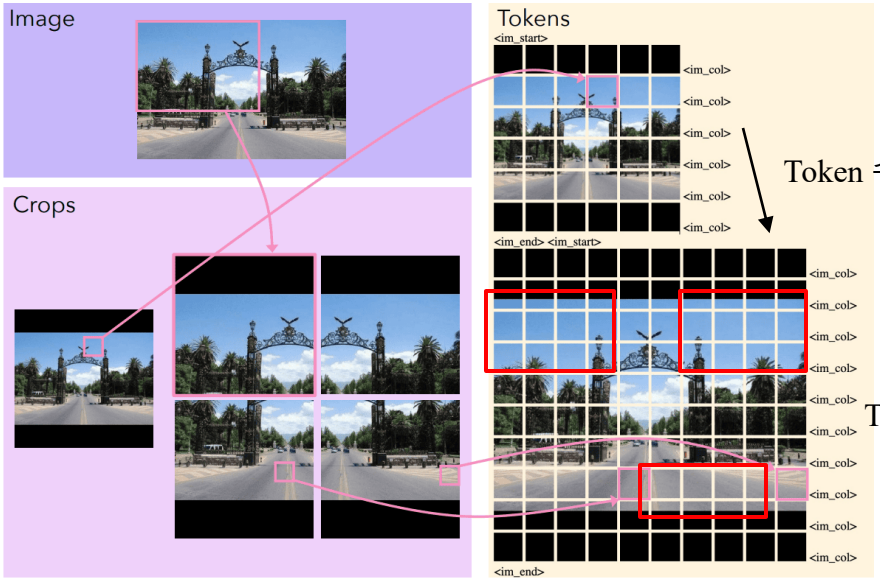
Background

- VLM구조



Background

- VLM 경량화
  - 필요성
    - 비디오 또는 높은 성능을 위해 해상도 높은 이미지를 사용하면 많은 수의 시각 토큰이 생김
    - 시각 토큰은 텍스트 토큰에 비해 중복도가 매우 높음
      - 높은 중복성에 의해 불필요한 token들도 LLM으로 들어가게 되고 이는 효율을 떨어뜨림



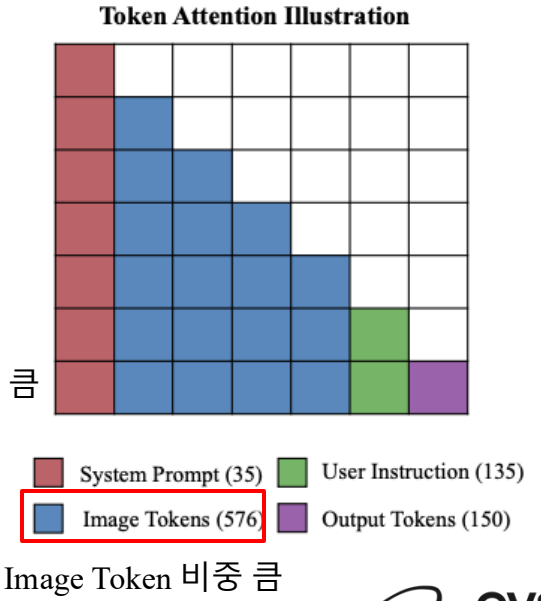


Image Token 비중 큼

Outline

- [CLS] Attention is All You Need for Training-Free Visual Token Pruning: Make VLM Inference Faster
- VScan: Rethinking Visual Token Reduction for Efficient Large Vision-Language Models

[CLS] Attention is All You Need for Training-Free Visual Token Pruning: Make VLM Inference Faster

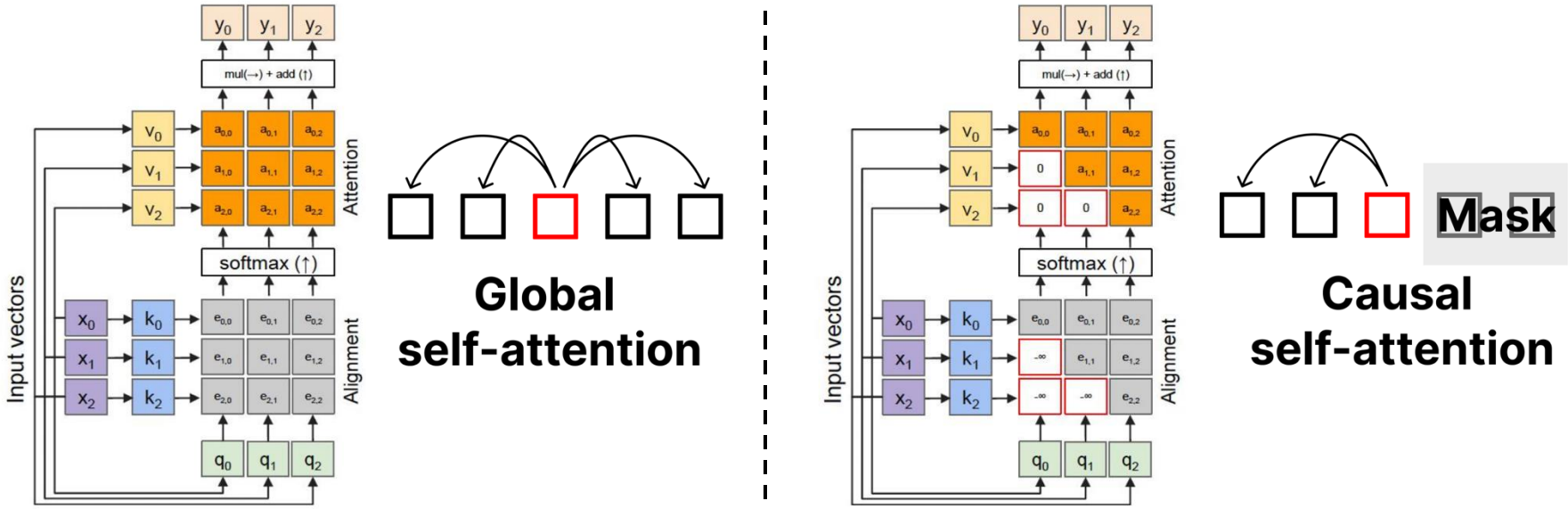
Introduction

- Visual 토큰을 줄이기 위한 기존 pruning 기법
  - LLM의 text-visual attention을 이용한 기법의 문제점
    - Pruning 비율이 커질수록 성능 크게 저하됨
      - 모델이 global정보를 artifact에 저장하기 때문에 global 토큰은 VLM 성능에 큰 영향을 미침
        - ✓높은pruning 비율에서는 text-visual attention의 부정확성이 이러한 artifact들을 식별하는데 어려움을 겪으면서 성능 저하



Preliminary

- VLM의 attention
  - Vision Encoder와 LLM Decoder의 attention 비교



Global self-attention
@Vision Encoder

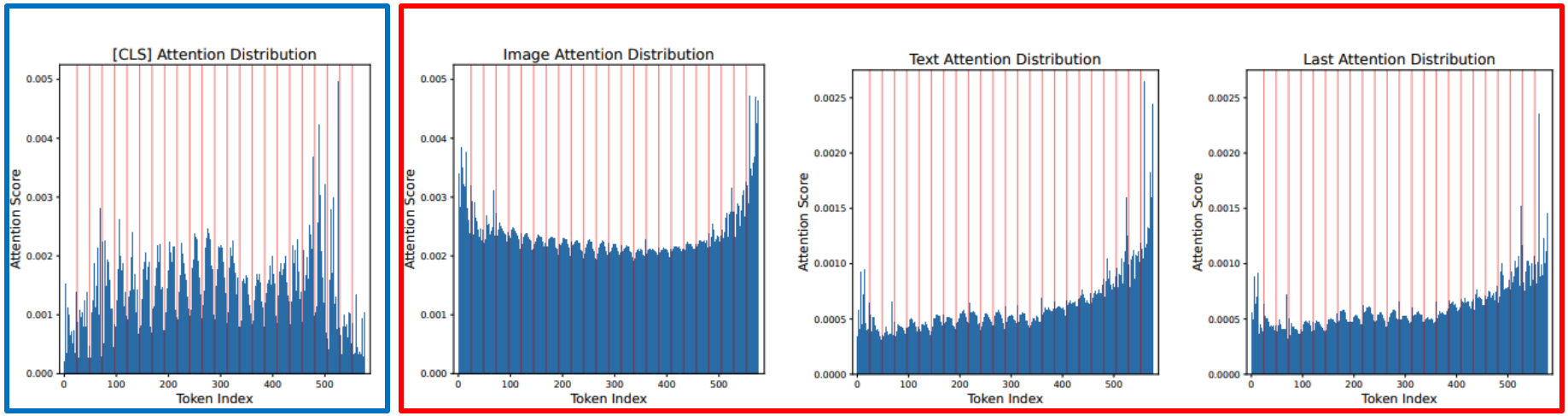
Causal self-attention
@LLM Decoder

Preliminary

- VLM의 attention
  - LLM에서 visual 토큰이 받는 attention
    - Image attention
      - visual 토큰이 다른 visual 토큰에게 준 attention
    - Text attention
      - Text 프롬프트(질문)이 visual 토큰에게 준 attention
    - Last attention
      - Output 생성을 위한 마지막 토큰이 visual 토큰에게 준 attention

Observation

- Inaccurate Text-Visual Attention in VLMs
  - Text-visual attention shift 현상
    - LLM decoder 내의 attention이 visual 토큰 시퀀스의 뒷부분에 더 집중하는 현상
      - LLM 디코더의 모든 attention 분포에서 토큰 인덱스가 클수록 attention점수가 높아지는 뚜렷한 경향을 보임

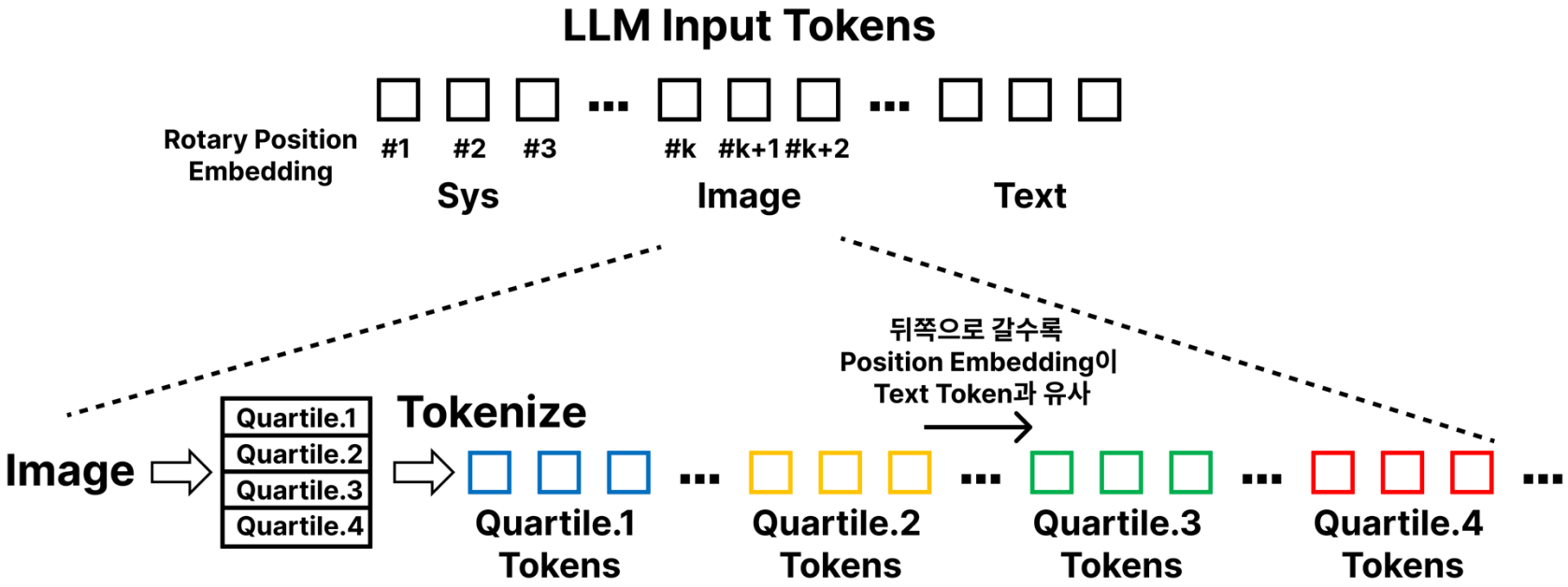


@Vision Encoder

@LLM Decoder

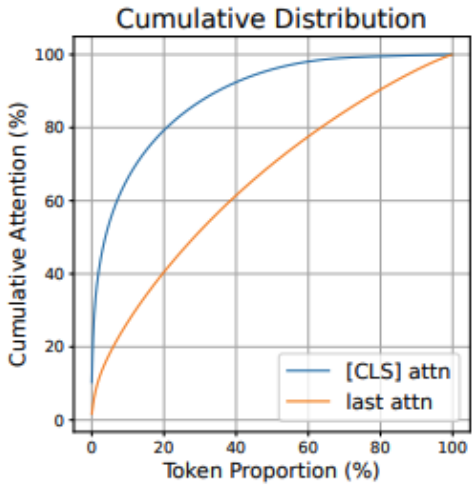
Observation

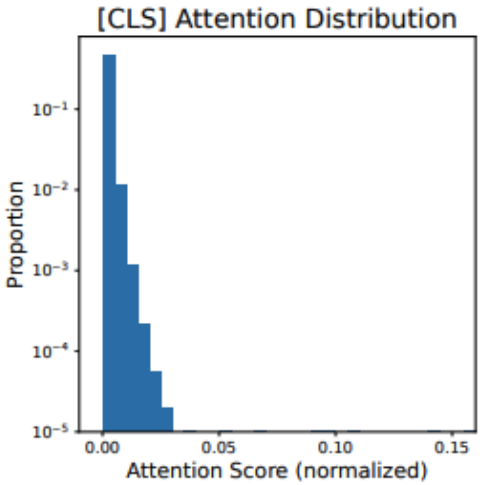
- Inaccurate Text-Visual Attention in VLMs
  - Text-visual attention shift 현상
    - 이는 LLM 내의 단방향 정보 흐름과 토큰 간의 상대적 위치에 기인
      - 뒤쪽 토큰은 앞쪽 토큰의 정보를 가지지만 앞쪽 토큰은 뒤쪽 토큰의 정보를 가질 수 없음
        - ✓Causal attention에서 앞쪽 정보까지 다 가지고 있는 뒤쪽 토큰에 더 attention을 주는 경향
      - Position embedding 또한 뒤쪽 토큰이 text 토큰과 더 유사

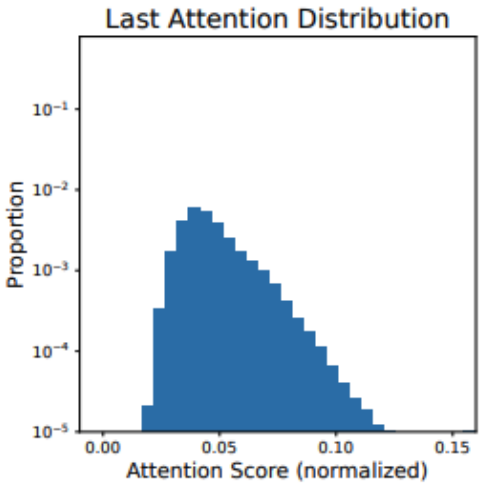


Observation

- Inaccurate Text-Visual Attention in VLMs
  - Text-visual attention dispersion 현상
    - Vision encoder에서의 attention분포에 비해 LLM 내의 attention분포는 덜 집중되고 퍼져 있음
      - Attention 분포의 집중도는 소수의 토큰이 대부분의 attention을 받고 나머지 대부분의 토큰이 낮은 점수를 받는 것을 의미
        - ✓집중도가 높을수록 모델이 중요한 토큰을 확신을 가지고 식별함을 의미

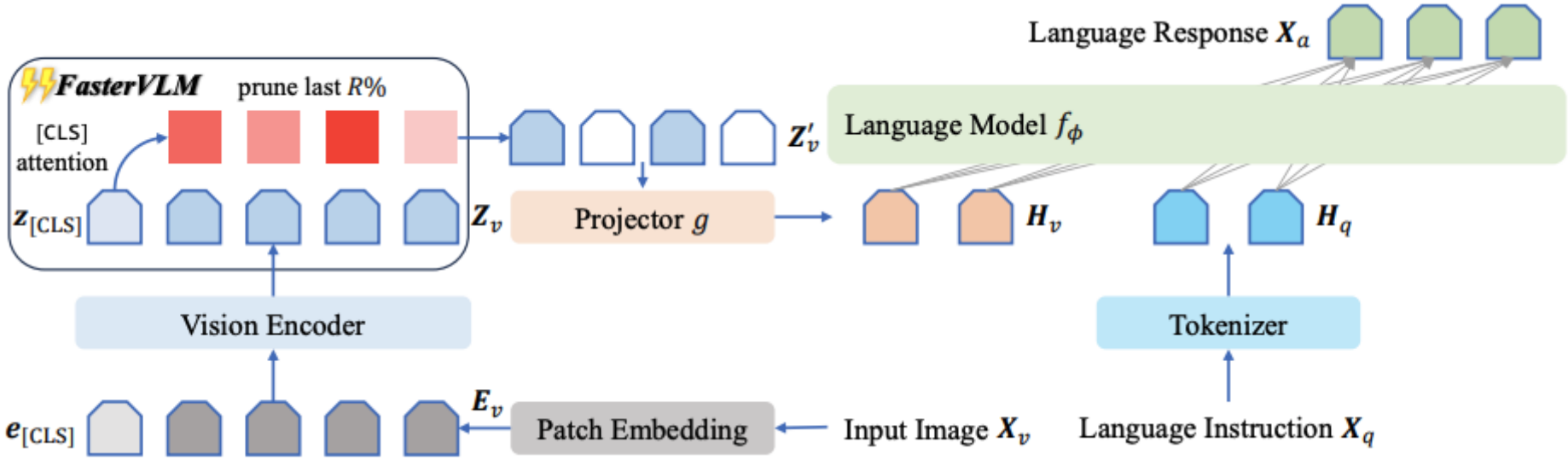






Method

- FasterVLM
  - Vision encoder 내 [CLS] 토큰의 attention을 활용하여 각 시각 토큰의 중요도 평가
    - Vision encoder의 마지막에서 두번째 레이어의 [CLS] 어텐션 스코어 이용해 연산 예산 R에 기반한 동적 임계값 τ를 설정
      - 이 임계값보다 낮은 토큰들은 제거
    - LLM이 처리하기 전에 낮은 attention 점수를 가진 토큰을 제거함으로써 최대한의 추론 속도 향상을 달성



Experiments

- Experimental Setup
  - Benchmarks
    - Image-based multi-modal benchmarks
      - VQAv2, GQA, VizWiz, ScienceQA-IMG, TextVQA, POPE , MME, MMBench, MMBench-CN, MM-Vet
    - Video question answering benchmarks
      - TGIF-QA, MSVD-QA, MSRVTT-QA, ActivityNet-QA
  - Models
    - LLaVA-1.5-7B/13B, LLaVA-NeXT-7B, Video-LLaVA-7B
  - Comparison methods
    - LLM decoder의 text-visual attention 기반 pruning 기법
      - FastV, FitPrune, SparseVLM

Experiments

- Main results
  - LLaVA-1.5 모델에 FasterVLM을 적용하고 기존 방법들과 종합적인 비교
    - 모든 text-visual attention기반 방법들은 pruning 비율이 커질수록 모델 성능이 저하
      - 이는 text 기반 어텐션이 가장 중요한 시각 토큰을 식별하는 데 신뢰성이 낮음을 시사
      - 반면 FasterVLM은 이러한 성능 저하를 크게 완화

Method	Reduction Ratio	# Token	VQAv2	GQA	VizWiz	SQA-IMG	TextVQA	POPE	MME	MMB	MMB-CN	MM-Vet	Average
LLaVA-1.5-7B	0%	576	78.52	61.94	50.06	69.51	58.21	85.88	1506.47	64.69	58.08	31.30	100.00%
FastV	50%	288	77.67	60.05	50.53	68.96	58.25	82.45	1513.06	64.26	58.16	31.70	99.33%
FitPrune			78.41	61.70	50.04	69.16	58.26	85.37	1499.70	64.60	58.16	31.10	99.73%
SparseVLM			76.67	58.78	51.03	68.57	57.49	83.87	1458.79	63.14	56.87	31.50	98.52%
FasterVLM			77.86	60.64	50.45	68.37	57.90	86.20	1471.96	63.83	56.70	34.80	**100.12%**
FastV	75%	144	74.07	56.58	51.29	69.11	57.38	73.74	1463.39	64.00	57.22	28.60	95.80%
FitPrune			76.14	59.38	51.30	69.01	56.49	80.75	1472.86	63.92	57.65	28.40	97.22%
SparseVLM			72.76	55.11	51.46	69.36	55.99	77.57	1351.65	59.54	51.03	29.90	93.84%
FasterVLM			76.19	58.34	51.97	67.92	57.07	83.46	1433.76	62.54	57.13	34.20	**98.75%**
FastV	90%	58	65.38	51.20	51.84	69.81	54.75	57.30	1210.36	59.97	51.72	27.20	87.97%
FitPrune			62.76	49.96	50.85	68.22	50.35	53.81	1147.46	56.27	45.53	21.80	82.07%
SparseVLM			62.90	48.86	49.36	67.23	48.99	65.82	1030.61	49.05	35.40	18.60	78.13%
FasterVLM			71.92	54.91	53.01	68.91	55.28	75.85	1348.63	60.57	54.90	30.10	**94.24%**
FastV	95%	29	55.92	46.03	49.10	70.00	51.56	35.47	971.56	50.17	42.18	18.90	74.93%
FitPrune			52.39	43.60	48.61	68.32	46.75	31.17	855.21	39.69	29.98	18.00	67.64%
FasterVLM			66.75	51.51	52.67	69.56	53.09	67.24	1254.80	58.51	51.98	27.50	**89.41%**

Experiments

- Ablation study
  - 다양한 전략에 대해 비교
    - (b)모든 이미지 패치 간 평균 self-attention 기반 pruning /(c)[CLS] attention 기반 pruning
      - 이미지 패치 간의 평균 self-attention 보다 [CLS] attention을 이용한 pruning이 더 좋은 이유는 [CLS] 토큰이 더 많은 전역 정보를 포함하기 때문
    - (d) LLM 내부 레이어에서 pruning하되 [CLS] attention 이용
      - 성능은 (c)FasterVLM과 유사하지만 LLM 이전에 pruning하는 경우보다 추론 효율 낮음

Pruning Strategy	TextVQA	POPE	MME
LLaVA-1.5-7B	58.21	85.88	1506.47
(a) w/ randomly prune 90% before LLM	48.45	74.77	1227.07
(b) w/ prune 90% by patch attn before LLM	54.15	70.39	1278.05
(c) w/ prune 90% by [CLS] attn before LLM	**55.28**	**75.85**	**1348.63**
(d) w/ prune 90% by [CLS] attn at LLM layer 2	55.18	73.51	1338.43
(e) w/ prune 90% by [CLS] attn and merge	49.72	68.19	1207.31

Experiments

- Ablation study
  - 다양한 전략에 대해 비교
    - (e)[CLS] attention으로 중요 토큰 선택 후 각 선택된 토큰을 key similarity가 높은 k개의 토큰과 병합
      - 병합을 많이 할수록 성능이 떨어짐

Pruning Strategy	TextVQA	POPE	MME
LLaVA-1.5-7B	58.21	85.88	1506.47
(a) w/ randomly prune 90% before LLM	48.45	74.77	1227.07
(b) w/ prune 90% by patch attn before LLM	54.15	70.39	1278.05
(c) w/ prune 90% by [CLS] attn before LLM	**55.28**	**75.85**	**1348.63**
(d) w/ prune 90% by [CLS] attn at LLM layer 2	55.18	73.51	1338.43
(e) w/ prune 90% by [CLS] attn and merge	49.72	68.19	1207.31

Experiments

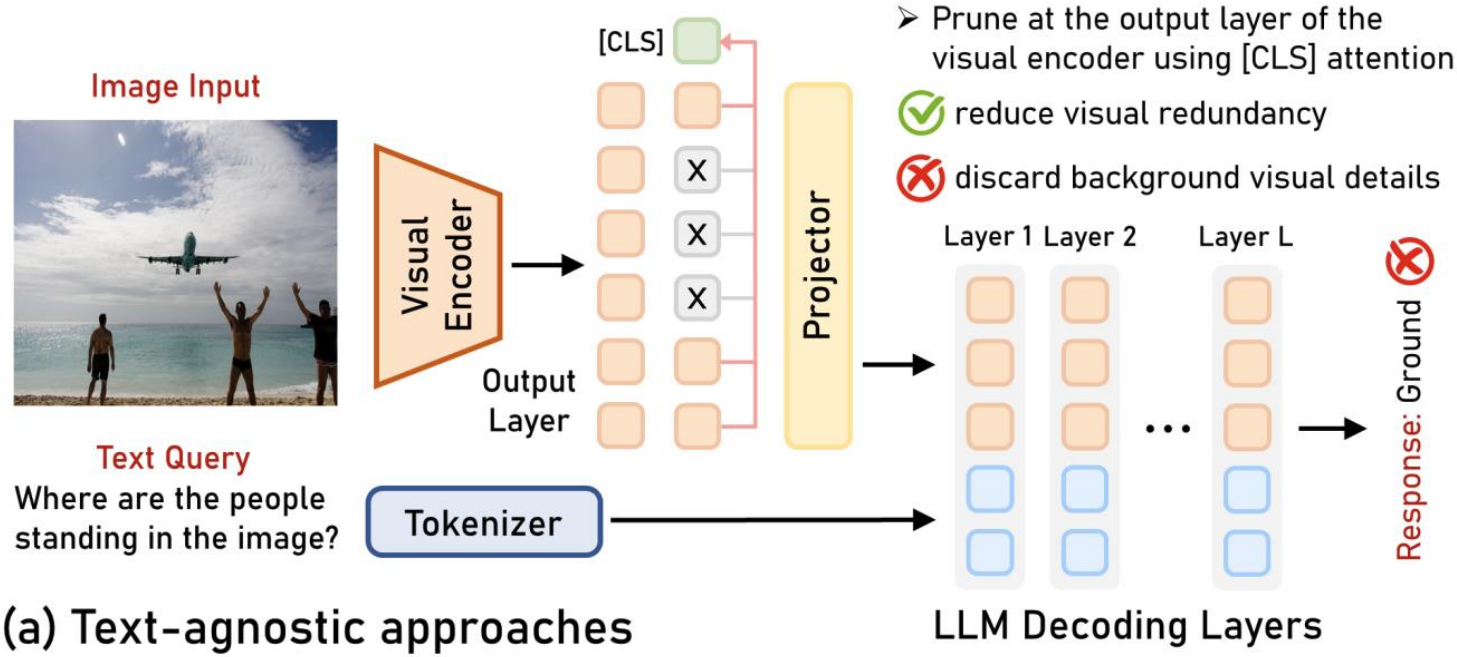
- Efficiency analysis
  - FasterVLM의 효율성 입증
    - LLaVA-NeXT-7B에서 FastV와 FLOPs, cache storage, CUDA 메모리 사용량, 추론 throughput을 비교 분석
      - 95% 축소 비율에서 FasterVLM은 LLaVA-NeXT에서 throughput을 **2.2배** 향상시킴
      - 동일한 축소 비율에서 FastV와 비교했을 때 더 적은 메모리를 사용하며 더 빠른 추론 속도를 달성
        - ✓이는 LLM 내부에서 pruning을 수행하면 KV cache 길이가 변동되어 현재 하드웨어와 호환성이 떨어지지만 FasterVLM은 이러한 문제 피함

Method	Reduction	# Token	FLOPs (T)	Storage (MB)	GPU Memory (GB)	Throughput (it/s)
LLaVA-NeXT-7B	0%	2880	41.65	1440	17.93	2.40
FastV	50%	1440	21.11	765	17.98	3.22
FasterVLM			**19.74**	**720**	**16.23**	**3.48**
FastV	95%	145	6.14	225	17.98	4.56
FasterVLM			**3.77**	**144**	**15.28**	**5.25**

VScan: Rethinking Visual Token Reduction for Efficient Large Vision-Language Models

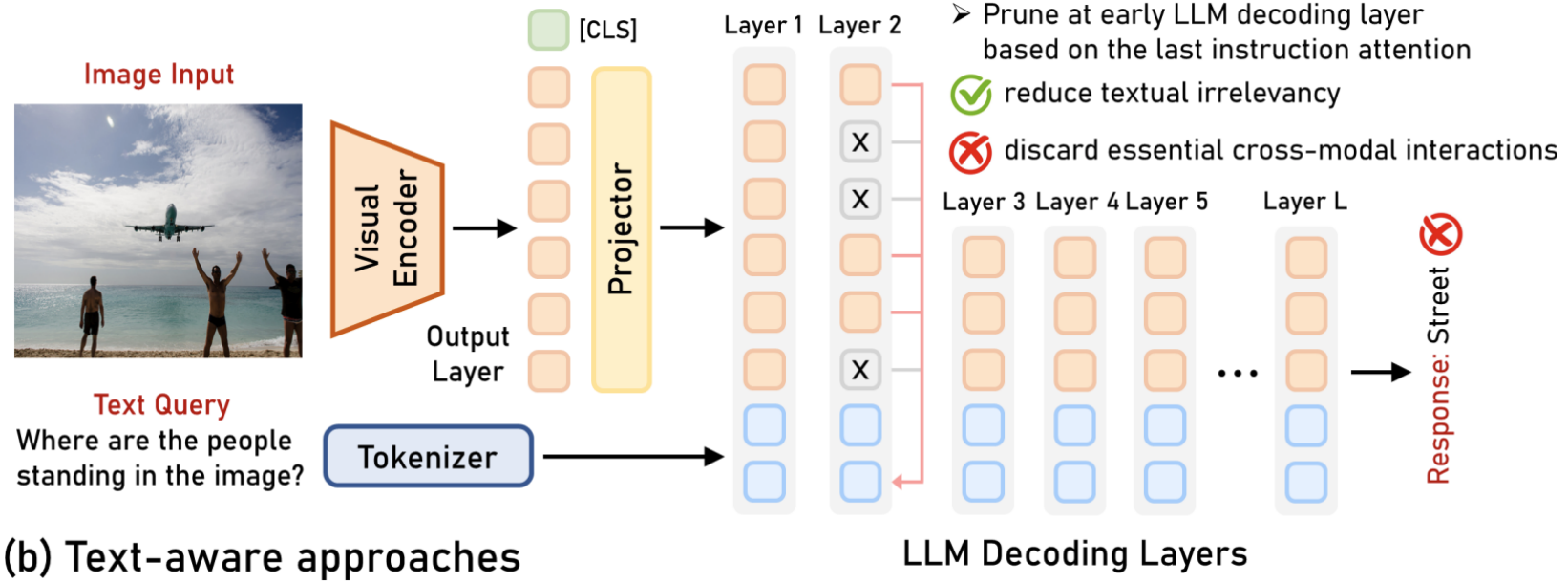
Introduction

- 기존의 visual token reduction 연구들
  - Text-agnostic pruning
    - Visual encoding 단계에서 토큰의 중요성과 고유성을 기준으로 시각적으로 중복된 토큰을 제거
      - 주로 Visual encoder 출력층의 [CLS] attention을 활용



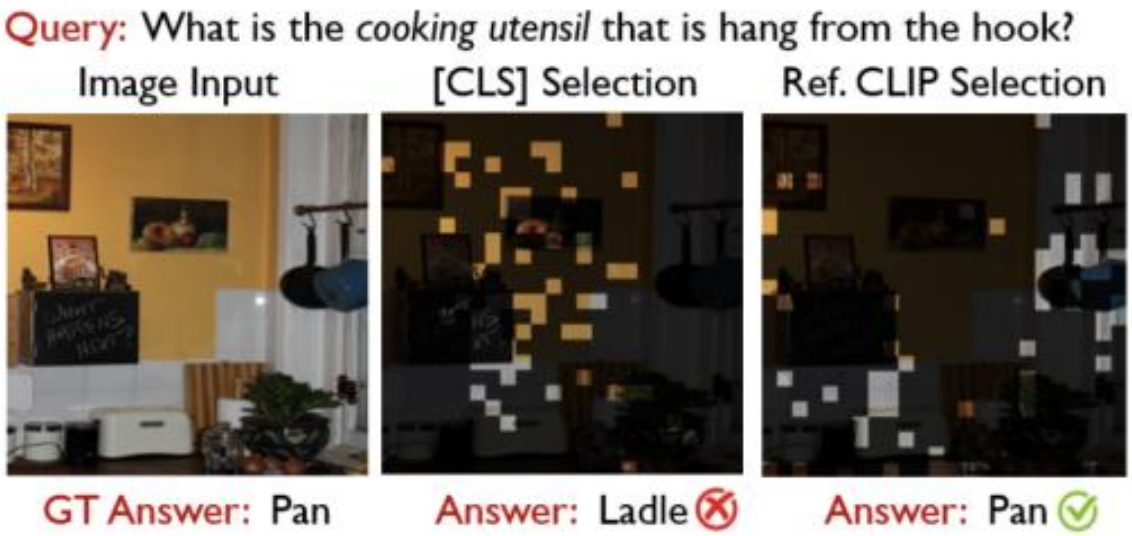
Introduction

- 기존의 visual token reduction 연구들
  - Text-aware pruning
    - Language Decoding 단계의 초기 레이어에서 text 쿼리와의 관련성이 낮은 토큰을 선별적으로 제거하여 task-specific한 정보를 보존



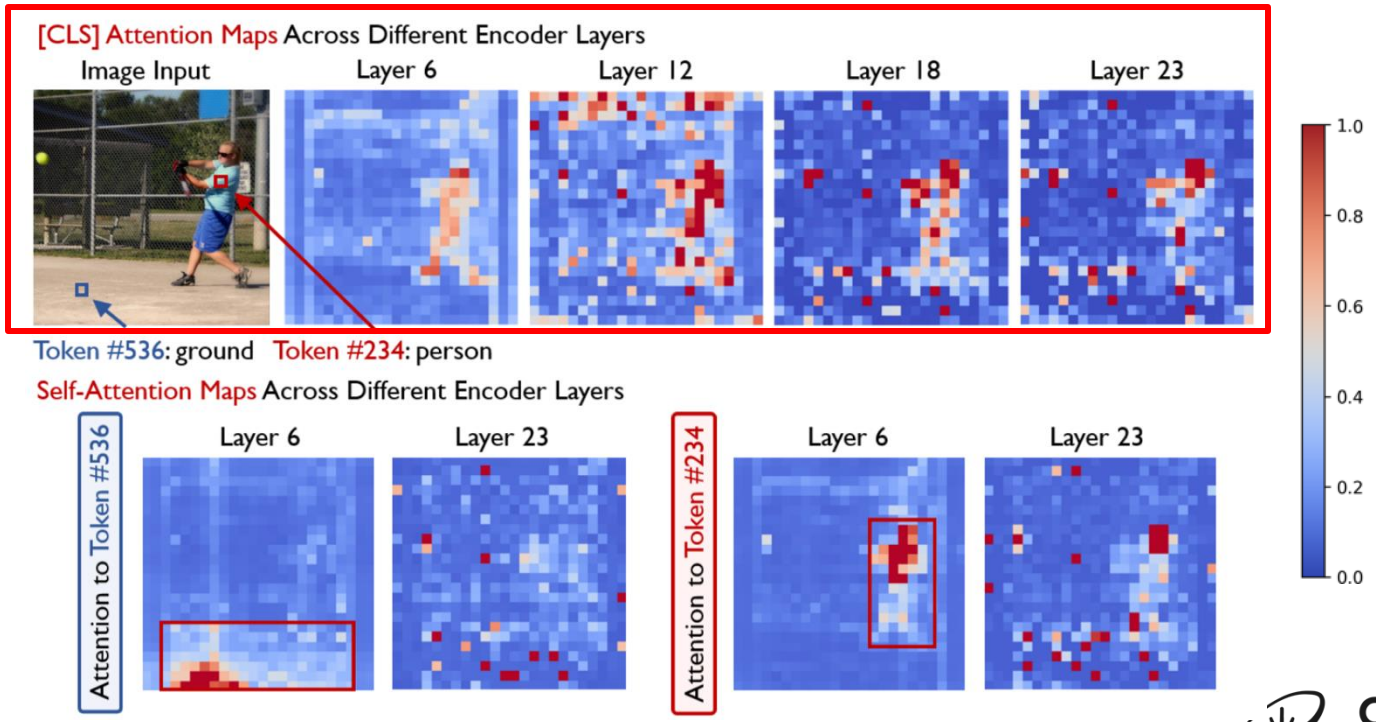
Introduction

- 기존의 visual token reduction 연구들
  - Limitations
    - Single-stage 설계와 LVLM 전체 파이프라인에서 시각 토큰이 어떻게 처리되고 활용되는지에 대한 체계적인 이해 부족으로 인해 성능이 제한되는 경우가 많음
    - 출력층의 [CLS] attention으로 선택된 토큰만으로는 모든 task 관련 시각 정보를 충분히 포착할 수 없음
      - 태스크를 수행하는데 중요한 의미정보를 담고 있을 수 있는 배경의 세부 정보를 과감하게 버림



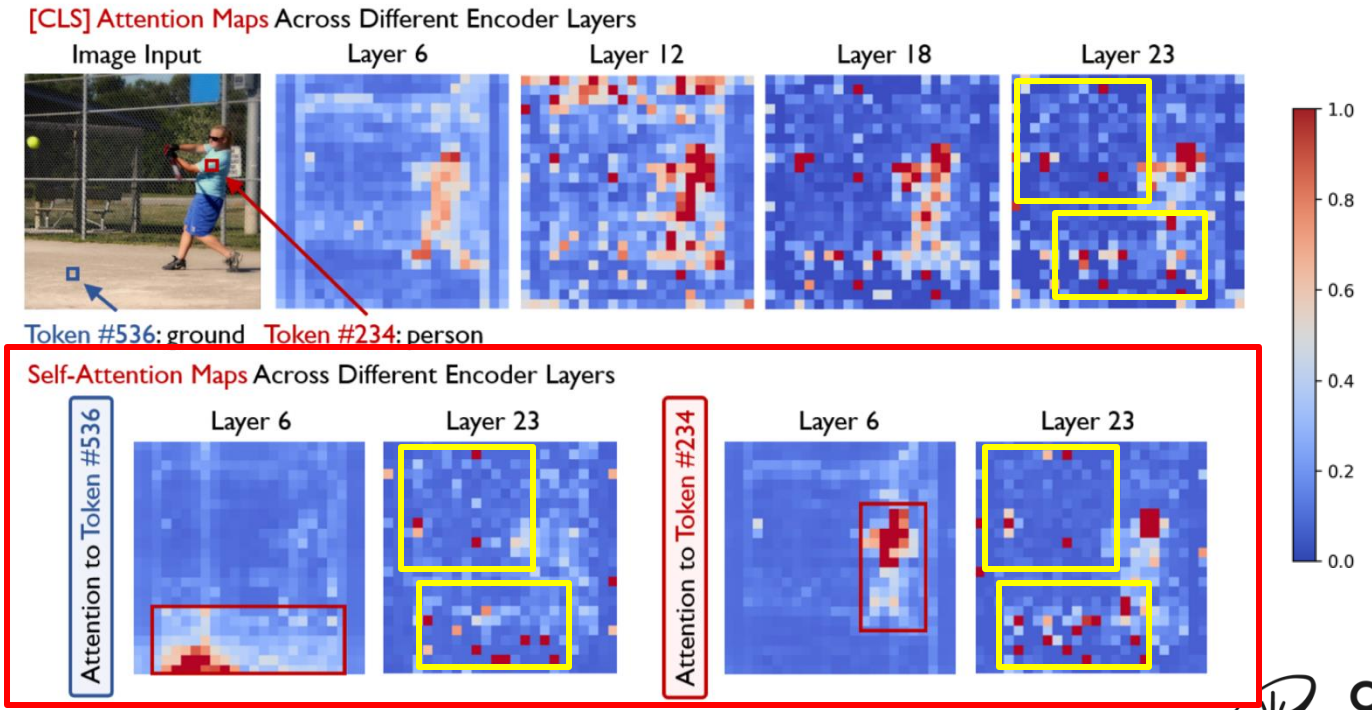
Empirical Analysis

- Key observations
  - Visual encoding stage
    - [CLS] attention
      - 얕은 층에서는 이미지 전반의 세밀한 local 정보를 포착
      - 깊은 레이어로 갈수록 전역적인 의미적 관련성을 반영한 주요 토큰에 집중
        - ✓Local => global의 경향 보임



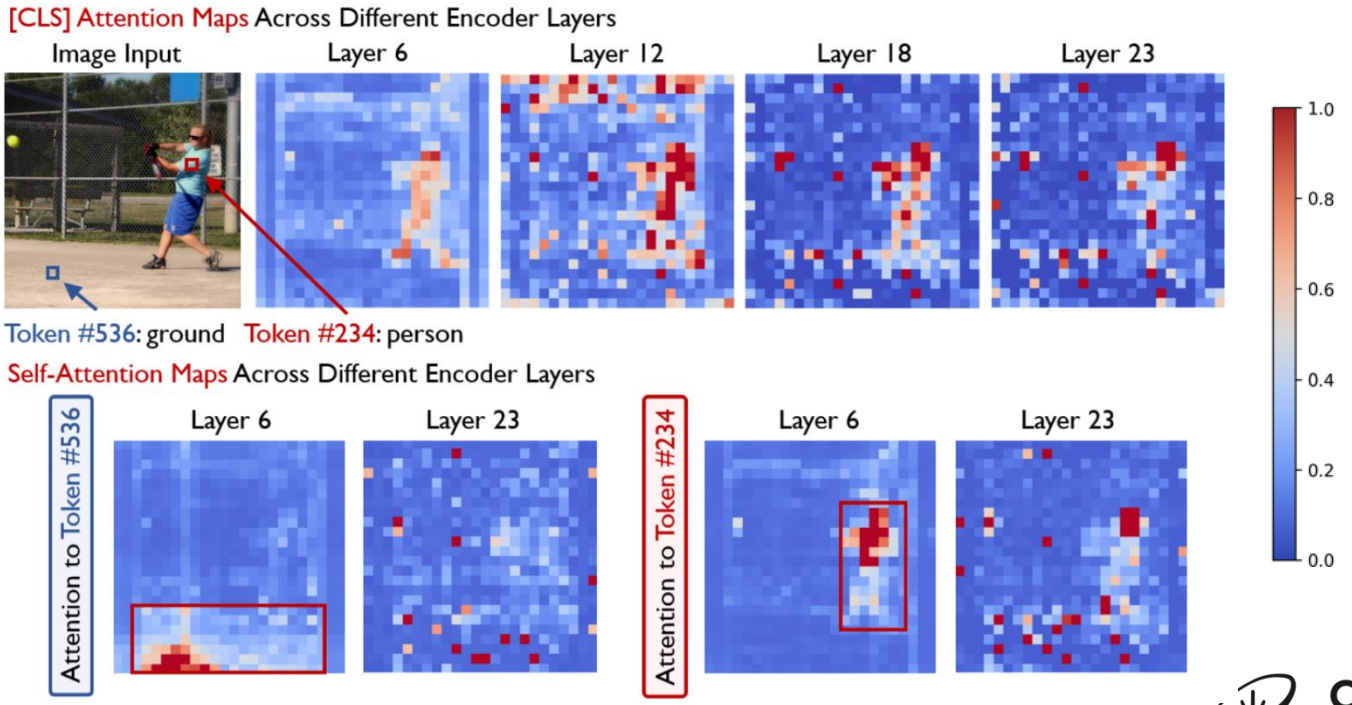
Empirical Analysis

- Key observations
  - Visual encoding stage
    - Self-attention
      - 얕은 층에서는 의미적으로 유사한 인접 영역에 집중
      - 깊은 레이어로 갈수록 attention이 인접 영역을 넘어 전체 이미지로부터의 문맥을 통합
        - ✓Local => global의 경향 보임



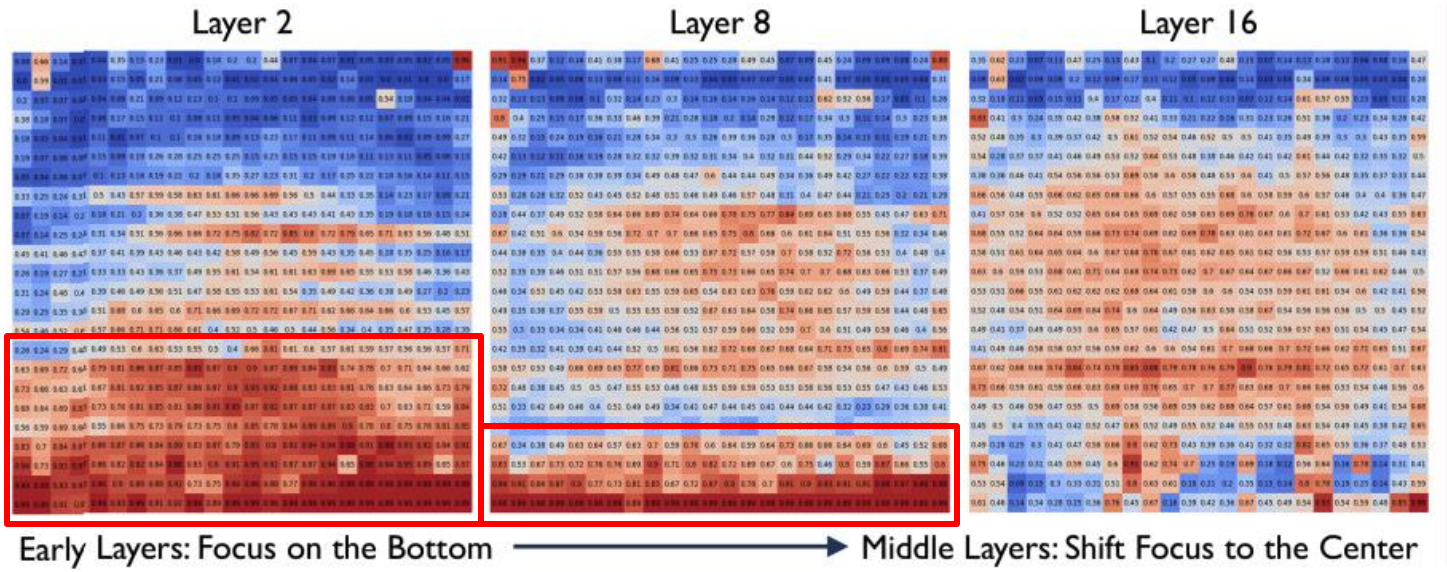
Empirical Analysis

- Key observations
  - Visual encoding stage
    - Visual encoder의 시각 정보 처리는 low-level의 local한 세부 정보를 포착하는 단계에서 high-level의 전역적 의미를 모델링하는 단계로 점진적으로 전환
    - 이는 출력층의 정보에만 의존하면 얕은 층에서의 local 정보를 간과할 수 있음을 보여줌



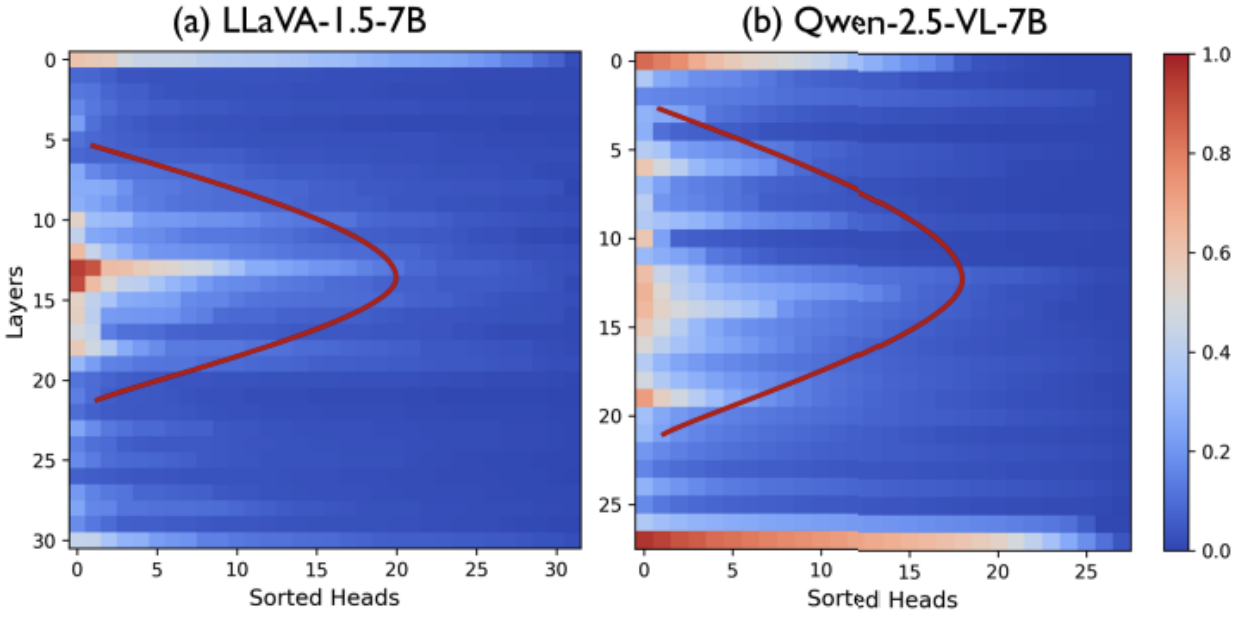
Empirical Analysis

- Key observations
  - LLM decoding stage
    - 토큰 선택에서의 위치편향
      - 마지막 instruction 토큰이 주로 인접 토큰에 주의를 집중하는 것
        - ✓일반적으로 가장 많은 정보와 task 관련 정보는 중앙에 있음에도 이미지 하단의 토큰을 선택하는 경향
        - ✓초기 레이어에서 뚜렷하게 나타나며 레이어가 깊어질수록 위치 편향이 줄어들고 어텐션이 중앙으로 이동



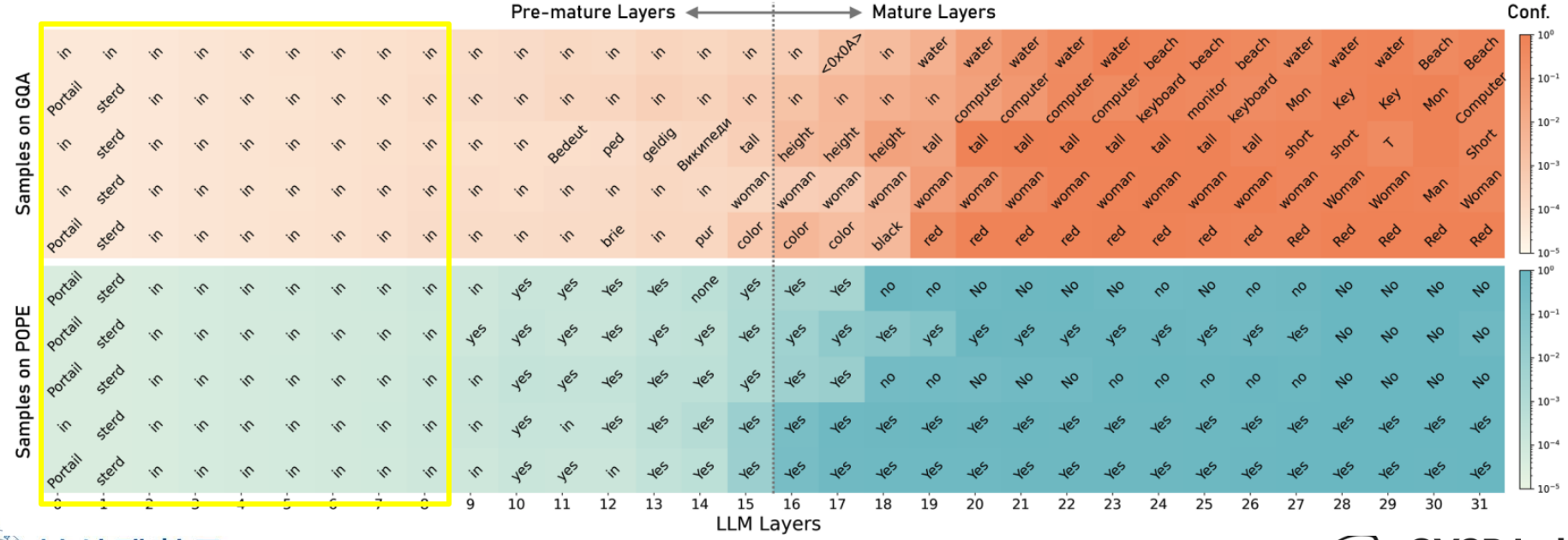
Empirical Analysis

- Key observations
  - LLM decoding stage
    - LLM이 어느 레이어부터 시각 정보를 수집하고 처리하는가
      - 중간 레이어가 visual 토큰과 주된 상호작용을 담당하고 초기 및 깊은 레이어는 주로 텍스트 정보 처리에 집중



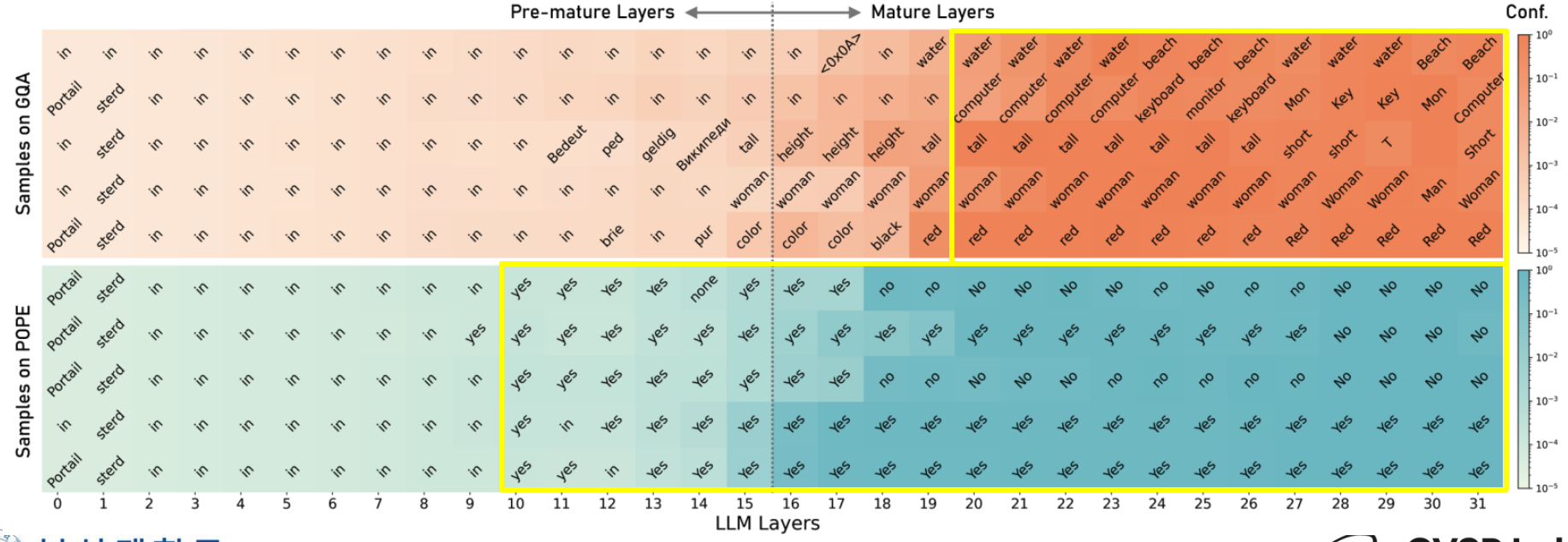
Empirical Analysis

- Key observations
  - LLM decoding stage
    - -LLM의 어느 레이어부터 다음 토큰 예측이 수렴하는가
      - 각 LLM 레이어의 hidden state를 linear projection 레이어에 입력하여 vocabulary logits과 다음 토큰에 대한 예측을 얻음
        - ✓초반 레이어의 hidden state는 아직 task를 위한 정보가 충분히 통합되지 않은 상태라 일반적인 다음 토큰 예측 패턴이 남아 불필요한 단어를 다음 토큰으로 출력



Empirical Analysis

- Key observations
  - LLM decoding stage
    - -LLM의 어느 레이어부터 다음 토큰 예측이 수렴하는가
      - 중후반 레이어부터 이미지 특징과 질문 문맥을 통합하는 cross-modal reasoning이 일어나면서 task에 맞는 정보가 hidden state에 반영되어 안정적인 출력
        - ✓다음 토큰 예측의 안정적인 수렴은 GQA와 같이 복잡한 open-ended task에서보다 POPE와 같은 예/아니오 task에서 빠르게 일어남

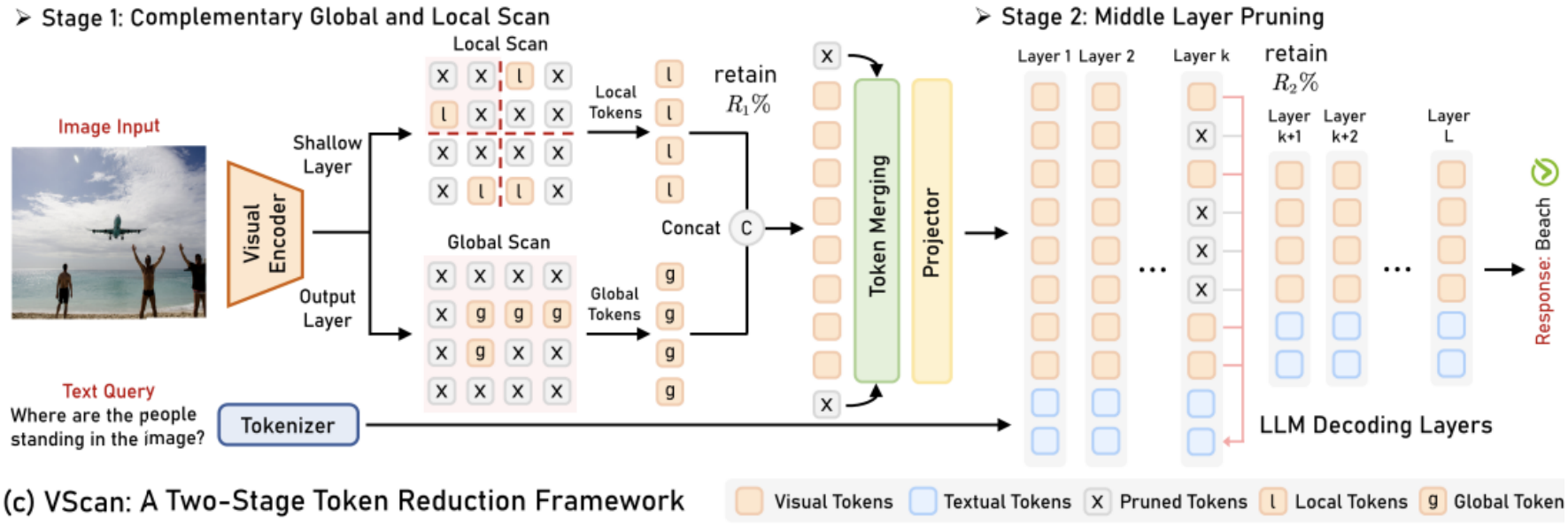


Empirical Analysis

- Key observations
  - 요약
    - Visual encoding stage
      - Visual encoder의 얕은 층에서는 지역적으로 중요한 토큰에 주목하며 세밀한 local 정보 중심으로 처리
      - 깊은 층에서는 점차 더 응축된 토큰 집합에 집중하며 global한 문맥을 포괄함
    - LLM decoding stage
      - 초기 레이어에서는 시퀀스 후반에 등장하는 visual 토큰에 대해 강한 위치적 편향 보임
      - 레이어가 깊어짐에 따라 모달 간의 상호작용이 나타나기 시작하며 보통 시각 정보가 언어 스트림에 효과적으로 통합되는 중후반 레이어에서 출력 토큰의 확률이 수렴함

Method

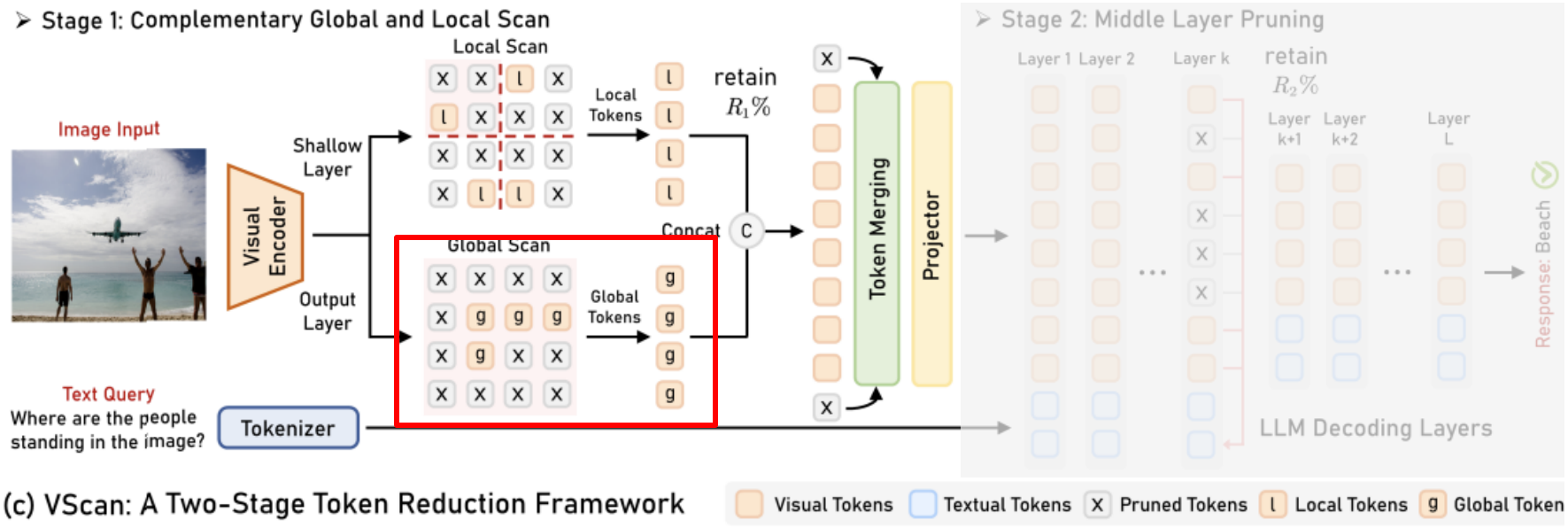
- VScan
  - 두 단계로 구성된 visual token reduction framework
    - 시각적 중복성과 텍스트 관련성 부족을 동시에 줄이기 위해 중요하지 않은 시각 토큰을 점진적으로 제거하는 두 단계의 학습 불필요 프레임워크 제안
      - Vision encoding 단계에서의 global과 local scan을 통합하여 토큰을 병합
      - LLM의 중간 레이어에서 pruning을 통한 중복 토큰 제거



(c) VScan: A Two-Stage Token Reduction Framework

Method

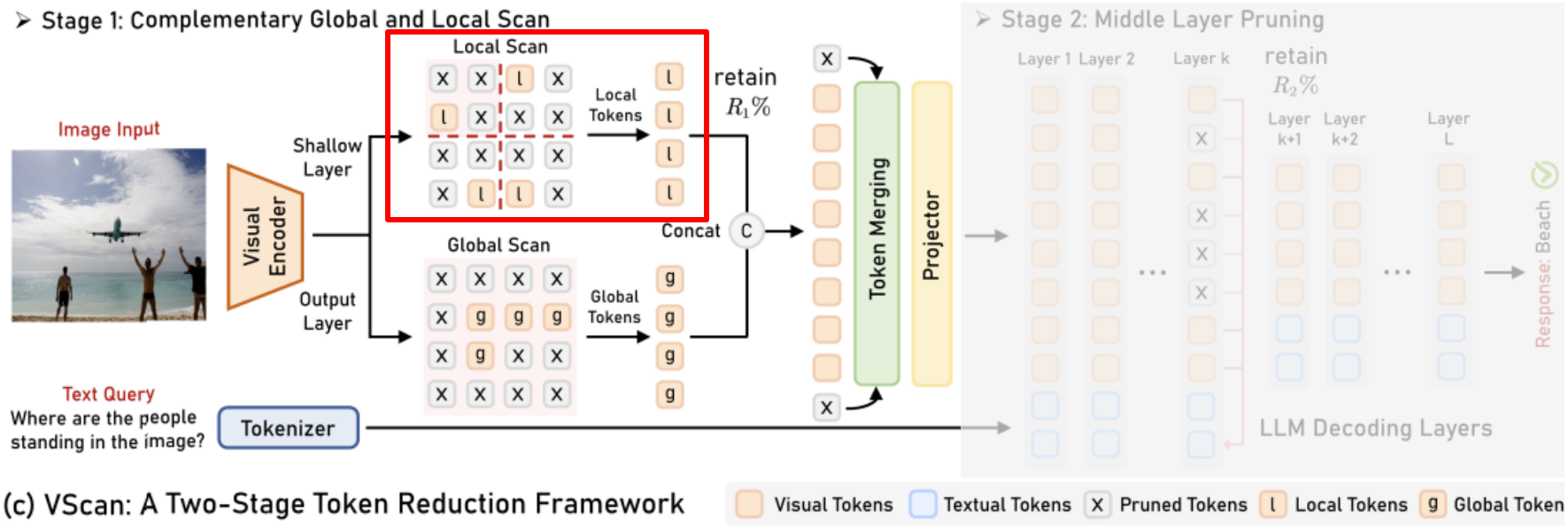
- VScan
  - Vision encoding 단계
    - Global scan
      - 최종 레이어가 global한 정보를 포착한다는 통찰을 고려하여 출력층에서 [CLS] 토큰으로 부터 가장 많은 어텐션을 받은 토큰을 global token으로 선택
        - ✓[CLS] 토큰이 없는 LVLM의 경우 각 visual 토큰이 다른 토큰들로부터 받는 self-attention의 평균값을 이용



(c) VScan: A Two-Stage Token Reduction Framework

Method

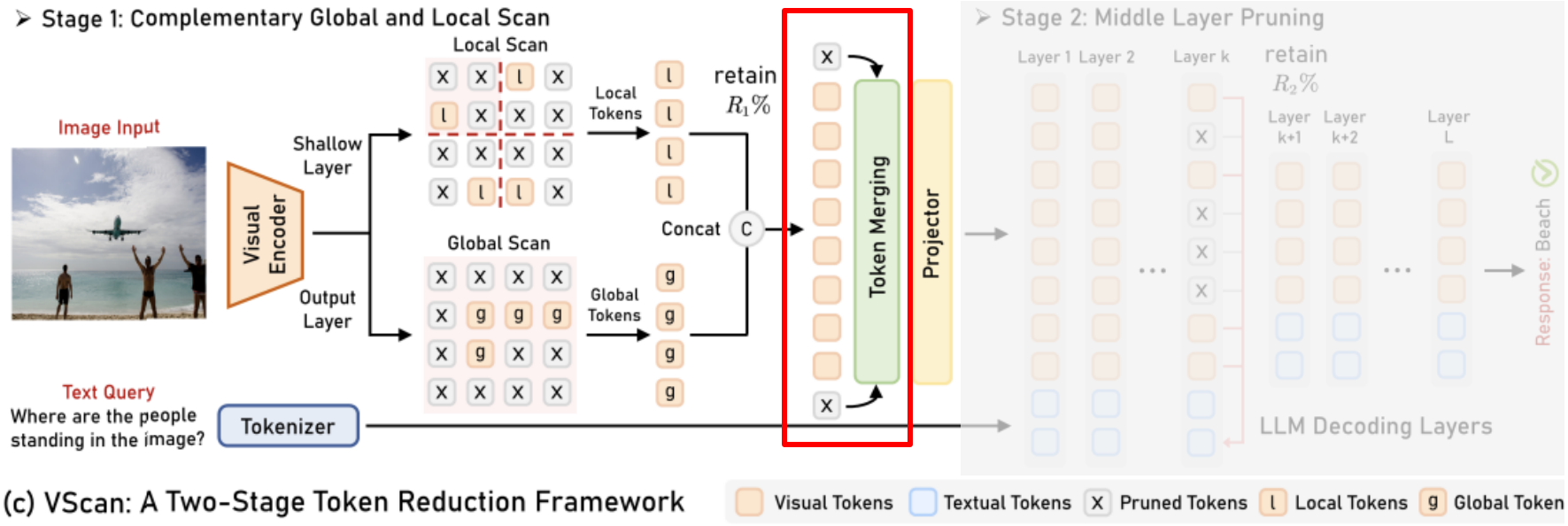
- VScan
  - Vision encoding 단계
    - Local scan
      - 세밀한 local 정보 포착을 위해 이미지를 겹치지 않는 윈도우로 나누고 각 윈도우에서 얕은 레이어의 [CLS]어텐션이 가장 높은 토큰 선택
        - ✓최종 선택된 토큰은 global 토큰과 local 토큰의 합집합으로 기존의 R1%의 비율을 남김
          - 기본적으로 두 토큰 집합의 비율은 일대일로 균형 맞춤



(c) VScan: A Two-Stage Token Reduction Framework

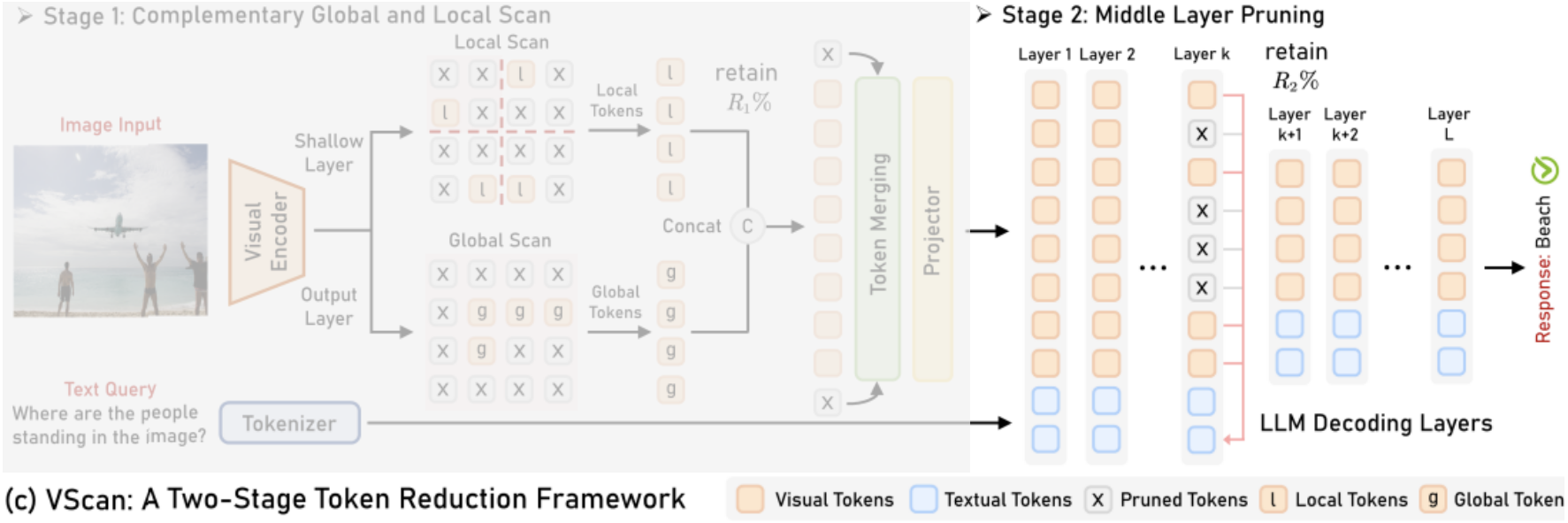
Method

- VScan
  - Vision encoding 단계
    - Token merging
      - 정보 손실을 완화하기 위해 선택되지 않은 토큰을 선택된 토큰 중 가장 유사한 토큰과 병합
        - ✓모든 선택되지 않은 토큰들이 코사인 유사도를 바탕으로 가장 가까운 선택된 토큰들에 할당되고 각 그룹 내에서 average merging



Method

- VScan
  - LLM decoding 단계
    - 위치 편향을 피하고 모달 간의 상호작용을 보존하기 위해 중간 레이어k에서 pruning
      - 마지막 instruction 토큰과의 어텐션을 계산하여 가장 높은 R2%의 visual 토큰을 선택



Method

- VScan
  - LLM decoding 단계 validation
    - 평균 토큰 축소율을 동일하게 유지하면서 pruning 레이어의 깊이의 영향 분석

평균 유지율 = $\frac{(\text{각 레이어에서 남은 토큰 수의 합})}{(\text{레이어 수} \times \text{초기 토큰 수})}$

✓LLM decoding 단계에서 평균 유지율이 75%가 되도록 각 레이어별로 R2설정하여 비교

중간 레이어에서 pruning 했을 때 가장 좋은 성능 보임

Settings	$R_1 = 50\%$	$R_1 = 25\%$
$k = 2, R_2 = 73.3\%$	59.6	56.8
$k = 8, R_2 = 66.7\%$	59.6	57.2
$k = 12, R_2 = 60.0\%$	59.6	58.6
$k = 16, R_2 = 50.0\%$	**60.7**	**58.7**
$k = 20, R_2 = 33.3\%$	<u>60.6</u>	**58.7**
$k = 24, R_2 = 0.0\%$	60.2	58.4

Experiments

- Experimental Setup
  - Models
    - LLaVA-1.5-7B, LLaVA-NeXT-7B, Video-LLaVA-7B, Qwen-2.5-VL
  - Comparison methods
    - Similarity기반 token merging
      - ToMe
    - Text-aware pruning
      - FastV, SparseVLM, PyramidDrop
    - Text-agnostic pruning
      - HiRED, Visionzip

Experiments

- Experimental Setup
  - Benchmarks
    - Image-based multi-modal benchmarks
      - VQAv2, GQA, VizWiz, ScienceQA, TextVQA, POPE , MME, MMBench, MMBench-CN
    - Video question answering benchmarks
      - TGIF-QA, MSVD-QA, MSRVTT-QA, ActivityNet-QA
  - Local scan layer
    - LLaVA 계열 모델은 l=6, Qwen-2.5-VL 모델은 l=8에서 수행
  - LLM pruning layer
    - LLaVA 계열 모델은 k=16, Qwen-2.5-VL 모델은 k=14에서 수행
    - 기본적으로 LLM 중간 레이어에서의 유지율 R2는 33.3%로 고정하고 목표 평균 축소율을 달성하기 위해 이에 맞게 R1을 조정
      - Ex:평균 유지율을 11.1%로 맞추기 위해 R1 = 16.7%, R2 = 33.3%로 설정

Experiments

- Main results
  - LLaVA-1.5-7B에서의 성능 비교
    - 이미지당 128개 및 192개 토큰만 사용했을 때에도 원본 LLaVA-1.5의 성능을 거의 유지
      - 더 높은 축소율에서 본 접근 방식의 장점이 더욱 두드러짐

Method	Venue	GQA	MMB	MMB^{CN}	MME	POPE	SQA^{IMG}	VQA^{V2}	VQA^{Text}	VizWiz	Average
Upper Bound, 576 Tokens (100%), 3.817 TFLOPs											
LLaVA-1.5-7B [39]	*CVPR'24*	61.9	64.7	58.1	1862	85.9	69.5	78.5	58.2	50.0	100.0%
Retain 192 Tokens in Average (↓ 66.7%), ∼1.253 TFLOPs											
ToMe [7]	*ICLR'23*	54.3	60.5	-	1563	72.4	65.2	68.0	52.1	-	88.5%
FastV [12]	*ECCV'24*	52.7	61.2	57.0	1612	64.8	67.3	67.1	52.5	50.8	90.4%
SparseVLM [69]	*ICML'25*	57.6	62.5	53.7	1721	83.6	**69.1**	75.6	56.1	50.5	96.1%
PyramidDrop [60]	*CVPR'25*	57.3	63.3	56.8	1797	82.3	69.0	75.1	56.5	**51.1**	97.2%
VisionZip [62]	*CVPR'25*	59.3	63.0	-	1783	85.3	68.9	77.4	57.3	-	97.8%
VScan (Ours)	-	**60.6**	**63.9**	**57.4**	**1806**	**86.2**	68.6	**77.8**	**57.7**	50.4	**99.0%**
Retain 128 Tokens in Average (↓ 77.8%), ∼0.833 TFLOPs											
ToMe [7]	*ICLR'23*	52.4	53.3	-	1343	62.8	59.6	63.0	49.1	-	80.4%
FastV [12]	*ECCV'24*	49.6	56.1	56.4	1490	59.6	60.2	61.8	50.6	51.3	85.4%
SparseVLM [69]	*ICML'25*	56.0	60.0	51.1	1696	80.5	67.1	73.8	54.9	51.4	93.7%
PyramidDrop [60]	*CVPR'25*	57.1	61.6	56.6	1761	82.3	68.4	72.9	56.6	51.0	96.2%
VisionZip [62]	*CVPR'25*	57.6	62.0	-	1763	83.2	**68.9**	75.6	56.8	-	96.2%
VScan (Ours)	-	**59.8**	**63.0**	**58.0**	**1792**	**86.1**	**68.9**	**77.1**	**57.3**	**51.7**	**98.8%**
Retain 64 Tokens in Average (↓ 88.9%), ∼0.415 TFLOPs											
ToMe [7]	*ICLR'23*	48.6	43.7	-	1138	52.5	50.0	57.1	45.3	-	70.1%
FastV [12]	*ECCV'24*	46.1	48.0	52.7	1256	48.0	51.1	55.0	47.8	50.8	76.7%
SparseVLM [69]	*ICML'25*	52.7	56.2	46.1	1505	75.1	62.2	68.2	51.8	50.1	87.2%
PyramidDrop [60]	*CVPR'25*	47.5	58.8	50.5	1561	55.9	**69.2**	69.2	50.6	50.7	86.6%
VisionZip [62]	*CVPR'25*	55.1	60.1	-	1690	77.0	69.0	72.4	55.5	-	92.7%
VScan (Ours)	-	**58.3**	**62.1**	**55.7**	**1698**	**85.0**	69.1	**75.4**	**55.6**	**51.8**	**96.7%**

Experiments

- Ablation studies
  - 유지율 R1과 R2의 변화
    - Visual encoding 과정에서의 토큰 선택과 LLM decoding 과정에서의 pruning 중 어느 한 쪽에만 지나치게 의존하면 성능이 떨어짐
    - 점진적인 2단계의 pruning 전략(R1 = 16.7%, R2 = 33.3%)을 적용할 때 최고 성능 달성

(a) Retention rates R_1 and R_2.

R_1	R_2	GQA	MME
11.1%	100.0%	56.7	1651
13.3%	66.7%	57.1	1683
14.8%	50.0%	57.5	1676
16.7%	33.3%	**58.3**	1698
22.2%	0.0%	52.7	**1720**

(b) Global & local tokens.

Global	Local	GQA	MME
0%	100%	58.0	1681
25%	75%	58.1	1689
50%	50%	**58.3**	**1698**
75%	25%	57.4	1688
100%	0%	57.5	1665

Experiments

- Ablation studies
  - Impact of mixing global and local tokens
    - Global 토큰과 local 토큰을 다양한 비율로 혼합 선택했을 때의 영향 조사
    - 동일 비율로 선택했을 때 최고의 성능을 얻을 수 있음
      - Global 토큰과 local 토큰이 서로 보완적인 역할을 하며 풍부한 시각 정보를 포착하고 모델의 추론 능력 향상에 함께 기여

(a) Retention rates R_1 and R_2.

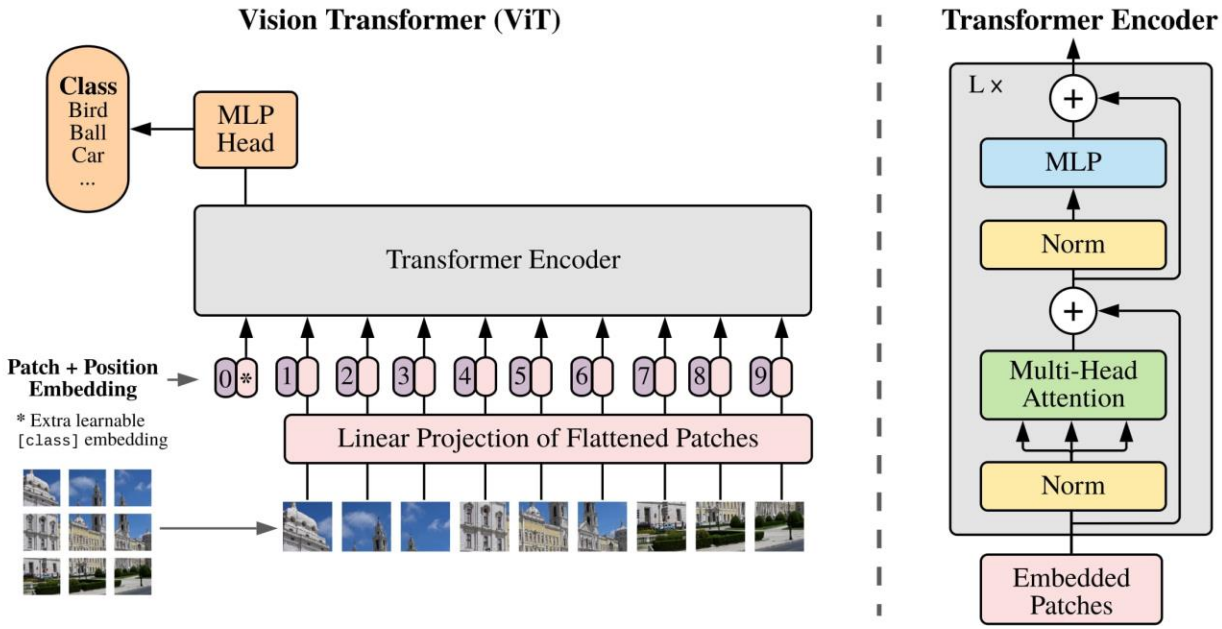
R_1	R_2	GQA	MME
11.1%	100.0%	56.7	1651
13.3%	66.7%	57.1	1683
14.8%	50.0%	57.5	1676
16.7%	33.3%	**58.3**	1698
22.2%	0.0%	52.7	**1720**

(b) Global & local tokens.

Global	Local	GQA	MME
0%	100%	58.0	1681
25%	75%	58.1	1689
50%	50%	**58.3**	**1698**
75%	25%	57.4	1688
100%	0%	57.5	1665

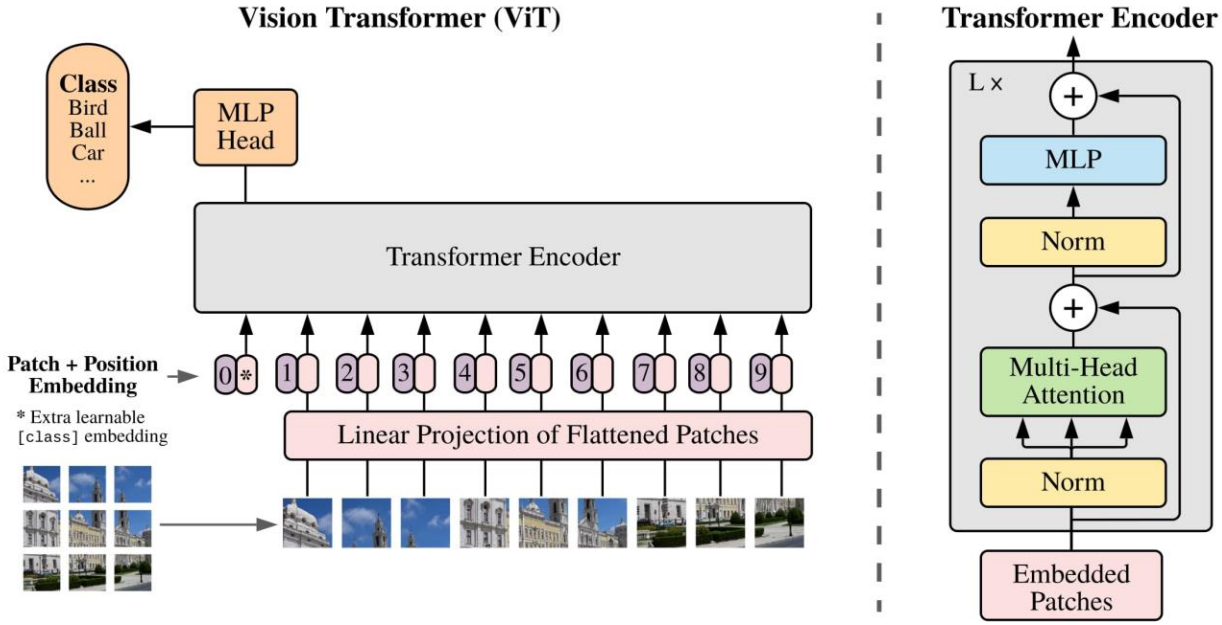
Appendix

- [CLS] token
  - BERT/ViT와 같은 encoder 기반 transformer에 쓰이는 클래스 분류용 요약 토큰
    - 입력 시퀀스의 맨 앞에 하나의 학습 가능한 벡터를 덧붙여 넣고 마지막 레이어의 [CLS]표현 하나만 분류헤드 (MLP)에 연결하여 전체 입력을 요약할 수 있는 토큰으로 사용
      - [CLS]자체는 토큰이지만 특정 단어나 (이미지)패치를 나타내진 않음
      - 다른 토큰들의 정보를 attention으로 끌어와 전체 요약을 만들도록 학습



Appendix

- [CLS] token
  - BERT/ViT와 같은 encoder 기반 transformer에 쓰이는 클래스 분류용 요약 토큰
    - 학습 신호 흐름
      - 분류 헤드는 오직 마지막 레이어의 [CLS]토큰의 표현을 읽어 Loss를 계산
        - ✓Gradient가 [CLS] → (해당 레이어의 어텐션 경로) → 다른 토큰들로 역류하면서 [CLS]가 유용한 정보가 있는 토큰들에 더 주의하도록 점점 학습
      - 결과적으로 얕은 층에선 로컬 디테일을, 깊은 층에선 전역/요약 정보를 더 많이 끌어오는 경향이 생긴다



Appendix

- 23번째 레이어에서 visual 토큰을 추출하는 이유
  - 논문 VisionZip 참고
    - 더 깊은 레이어로 갈수록 어텐션 정보가 소수의 지배적인 토큰에 집중되며 23번째(-2) 레이어에서 그 집중도가 절정에 이름
      - 마지막 레이어의 경우 어텐션이 더 분산됨
        - ✓이는 해당 토큰들이 CLIP의 text branch와 contrastive loss로 정렬되면서 원래의 이미지 표현을 충분히 반영하지 못할 수 있기 때문



Figure 5. Visualization of attention distribution across layers

Appendix

- 평균 유지율 계산
  - 평균 유지율 = $\frac{(\text{각 레이어에서 남은 토큰 수의 합})}{(\text{레이어 수} \times \text{초기 토큰 수})}$
    - LLM 디코딩 전체 구간에서 평균적으로 몇 % 토큰을 유지했는지 계산
    - Ex: k=2, R2=50%일때
      - Layer 1~2: **288개** 토큰 (50% 축소 전)
      - Layer 3~32: **144개** 토큰 (R2=50% 적용 후)
      - 평균 유지율 = $\frac{(2\times288)+(30\times144)}{(32\times576)} = 26.6\%$

Appendix

- Global 토큰과 local 토큰의 중복 처리
  - Shallow 레이어에서 local_indicies 를 뽑아 local token 들을 선택함
  - Deep 레이어에서 local_indicies에 해당하는 부분을 0으로 두고 global_indicies를 top_k로 뽑음
    - 따라서 두 토큰 집합을 구성하고 token_indices = cat(local, global)해도 인덱스가 겹치지 않도록 함
    - 이 방식은 local 토큰을 먼저 선택하고 global 토큰을 겹치는 부분 제외하고 선택하기 때문에 선택되는 global 토큰의 평균 스코어가 낮아짐
    - 또한 global 토큰을 먼저 선택할 때와 선택되는 최종 토큰에서 차이 발생

Appendix

- Benchmarks
  - VQAv2
    - 하나의 이미지에 대한 개방형 질문을 포함하는 데이터셋



Appendix

- Benchmarks
  - GQA
    - 현실세계의 시각적 추론과 조합적 질문응답을 위한 데이터셋

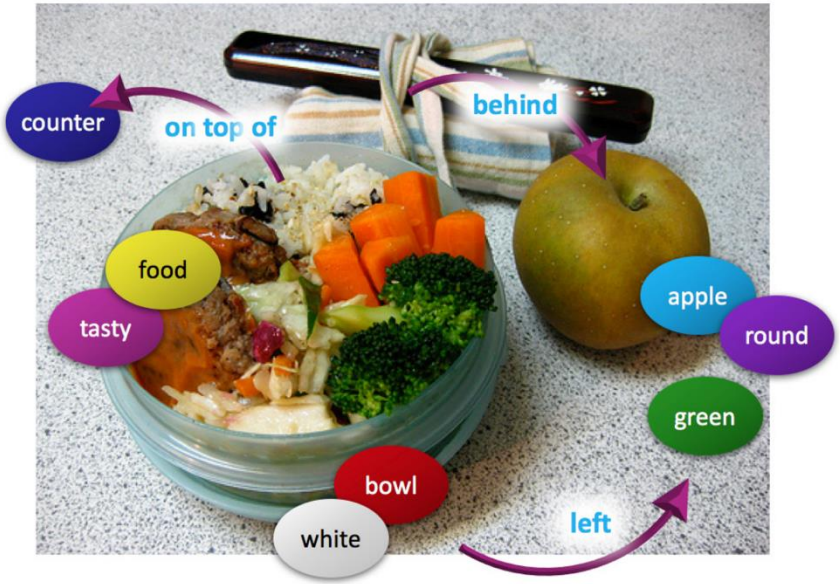


Figure 1: Examples from the new GQA dataset for visual reasoning and compositional question answering:
*Is the **bowl** to the right of the **green apple**?*
*What type of **fruit** in the image is **round**?*
*What color is the **fruit** on the right side, red or **green**?*
*Is there any **milk** in the **bowl** to the left of the **apple**?*

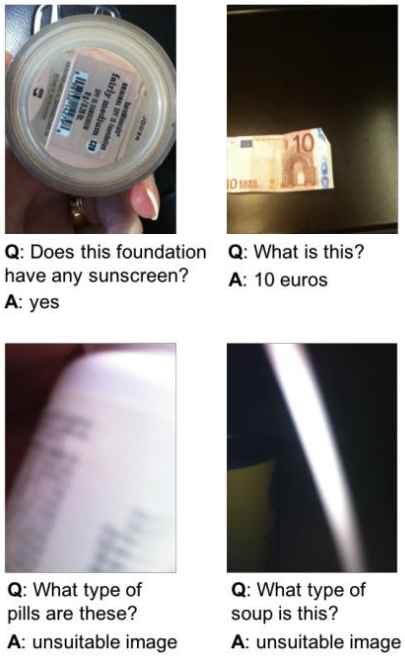


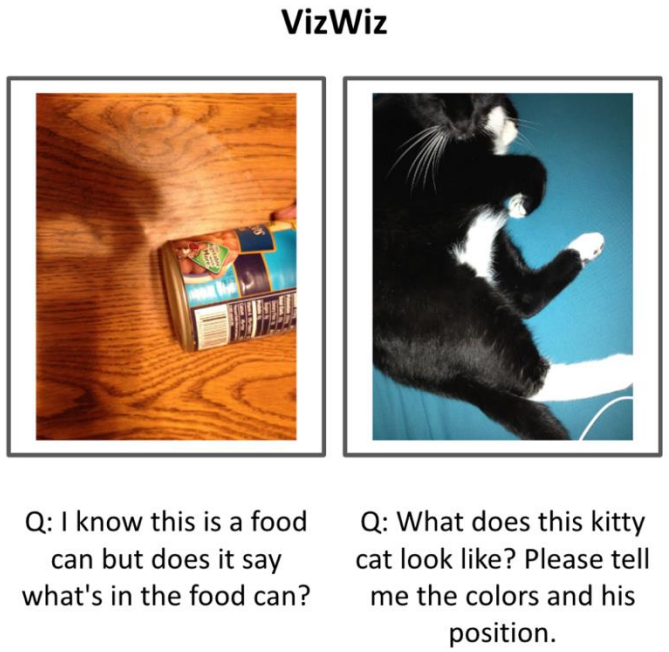


A1. Is the **tray** on top of the **table** black or light brown? light brown
A2. Are the **napkin** and the **cup** the same color? yes
A3. Is the small **table** both oval and wooden? yes
A4. Is there any **fruit** to the left of the **tray** the **cup** is on top of? yes
A5. Are there any **cups** to the left of the **tray** on top of the **table**? no
B1. What is the brown **animal** sitting inside of? **box**
B2. What is the large **container** made of? cardboard
B3. What **animal** is in the **box**? **bear**
B4. Is there a **bag** to the right of the green **door**? no
B5. Is there a **box** inside the plastic **bag**? no

Appendix

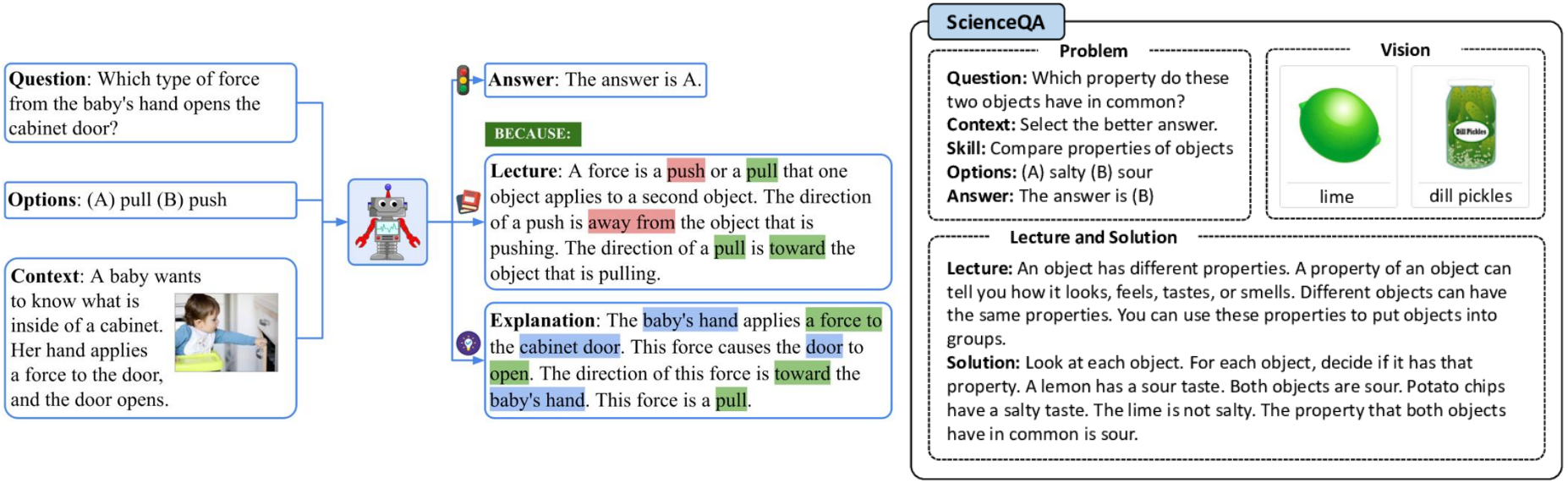
- Benchmarks
  - VisWiz
    - 자연스러운 실제 환경에서 생성된 목표 지향적 VQA데이터셋
      - 시각 장애인 사용자들이 휴대폰으로 사진을 촬영하고 그 이미지에 대해 음성으로 질문을 녹음한 시각적 질문으로 구성





Appendix

- Benchmarks
  - ScienceQA
    - 약 2.1만 개의 멀티모달 객관식 과학 질문으로 구성 각 질문에 대해 강의(lectures)와 설명(explanations)이 포함된 정답 주석(annotation)을 제공



Appendix

- Benchmarks
  - TextVQA
    - 모델이 이미지 속 텍스트를 읽고 추론하여 이에 대한 질문에 답해야 하는 과제를 요구

(a)



Q: what number is on the black and white sign?
OCR list: ['to', 'interstate', '15', 'to', '201', 'theu', 'garlicburger', '89']
Answer: 201

(b)



Q: what country in on the coin?
OCR list: ['of', 'america', 'states', 'tenham', 'epluribus', 'unum', 'half', 'dollar', '3/4', '92.', '9999', 'fins', 'gold']
Answer: united states of america



(a) Question: which processor Brand is featured on the top left?
Answer: intel



(b) Question: which brand are the crayons?
Answer: crayola



(c) Question: what is the name of the bose speaker style in these boxes?
Answer: freestyle



(d) Question: what is the license number?
Answer: cu58 ckk

Appendix

- Benchmarks
  - POPE(Polling-based Object Probing Evaluation)
    - 객체 환각을 보다 안정적이고 유연하게 평가하기 위해 설계된 벤치마크

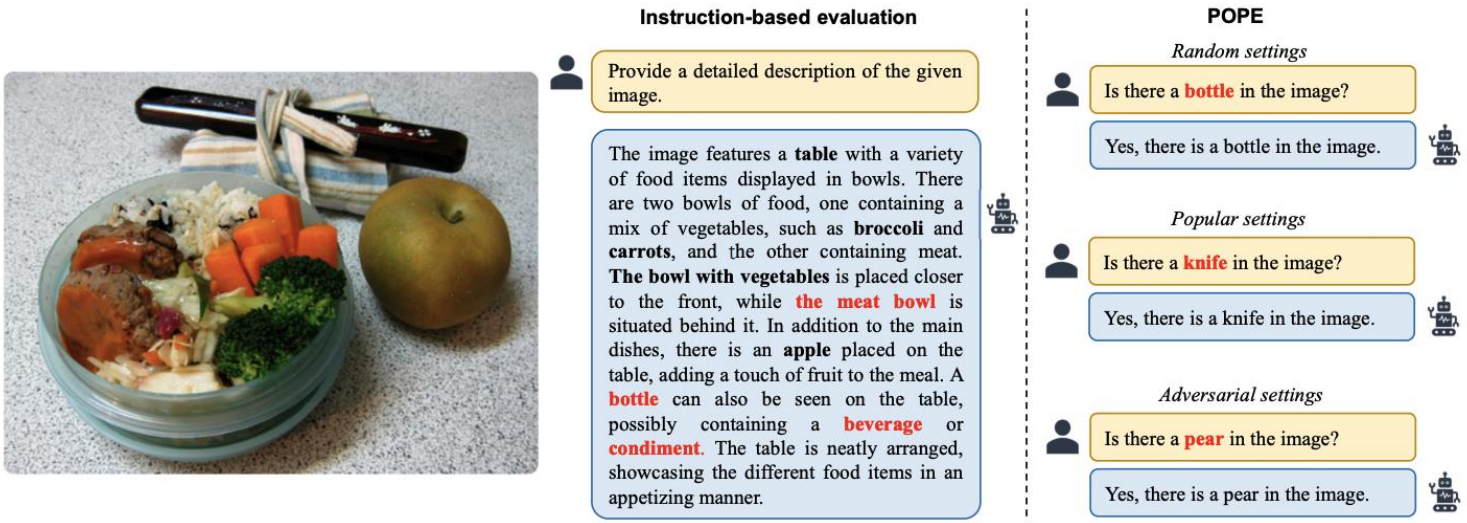


Figure 1: Cases of object hallucination in LVLMs. **Bold** objects are ground-truth objects in the annotations and **red** objects are hallucinated objects by LVLMs. The left case is from the traditional instruction-based evaluation method, and the right cases are from three variants of POPE.

Appendix

- Benchmarks
  - MMBench
    - 객체 위치 추론(object localization), 사회적 추론(social reasoning) 등 20개의 능력 차원을 포함하는 3,000개 이상의 객관식 문항을 포함

