2026.07.31 (Summer Seminar)

Object Counting Method & Visual Understanding

1. Generalized-Scale Object Counting with Gradual Query Aggregation [AAAI 2026]

2. Counting Circuits: Mechanistic Interpretability of Visual Reasoning in Large Vision-Language Models [arXiv 2026]

Sogang University
Vision & Display Systems Lab, Dept. . of Artificial Intelligence

Presented By
HeeDong Seo

1) J.Pelhan, A.Lukezic, M.Kristan , “Generalized-Scale Object Counting with Gradual Query Aggregation,”AAAI, 2026

2) L.Che, Z.Xue, Y.Quan, B.Liu, Z.Shi, M.Hurst, J.Feldman, R.Tang, R.Krishna, V.Pavlovic, “Counting Circuits: Mechanistic Interpretability of Visual Reasoning in Large Vision-Language Models,” arXiv preprint arXiv:2603.18523v1, 2026

Outline

- Object Counting Method
- Generalized-Scale Object Counting with Gradual Query Aggregation [AAAI 2026] [1)]
- Counting Circuits: Mechanistic Interpretability of Visual Reasoning in Large Vision-Language Models [arXiv 2026] [2)]

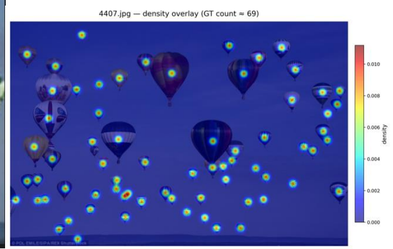
Object Counting Method

Detection-based vs Density Estimation-based Approaches

- Detection-based Approaches
  - 이미지 내 객체의 Bounding Box를 모두 추론한 뒤 그 개수를 합산하는 직관적인 방식
- Density Estimation-based Approaches
  - 개별 객체를 탐지하는 대신, 이미지 전체의 공간적 밀도 지도를 회귀방식으로 예측하고 이를 적분하여 총합을 구함

Detection-based (GT Point or GT Point & Bounding Box)

Density Estimation-based (Density Map)

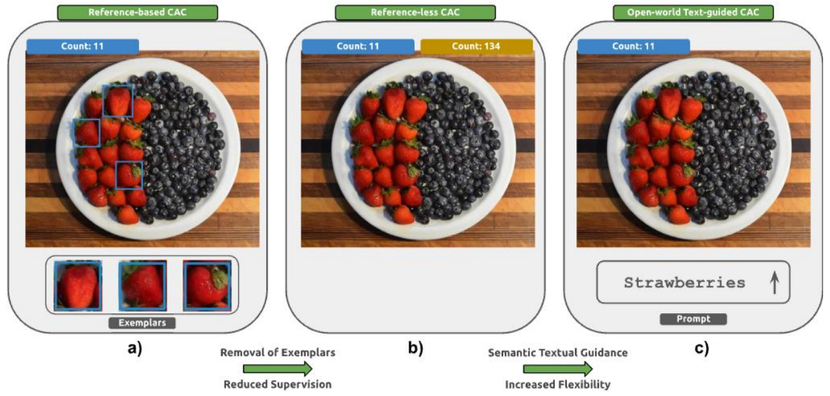


Density Map heatmap

1) L.Ciampi, A.Azmoudeh, E.E.Akbaba, E.Saritas, Z.A.Yazici, H.K.Ekenel, G.Amato, F.Falchi, “Computer Vision and Image Understanding,” CVIU, 2025

Class-Agnostic Counting (CAC)

- CAC : 훈련 중에 보지 못한 임의의 객체 범주까지 유연하게 셀 수 있는 접근법
  - Reference-based Approaches
    - 훈련 및 추론 시에 Counting할 객체의 시각적 프로토타입 역할을 하는 주석 처리된 바운딩 박스 exemplars를 입력받아 카운팅 수행
  - Reference-less Approaches
    - 사용자가 시각적 예시를 제공할 필요 없이, 이미지 내의 고유한 패턴과 self-similarity을 통해 반복되는 지배적인 객체 클래스를 모델이 자동으로 추론하여 세는 방식
  - Open-world Text-guided Approaches
    - 자연어 텍스트 프롬프트를 입력하여 세고자 하는 객체 클래스를 지정하는 가장 최신의 유연한 패러다임
      - 사람이 사물을 묘사하는 방식과 가장 유사하며, CLIP이나 GroundingDINO와 같은 대형 Vision-Language Foundation Models 을 활용하여 의미론적 제어 가능



class-agnostic counting paradigms[1)]

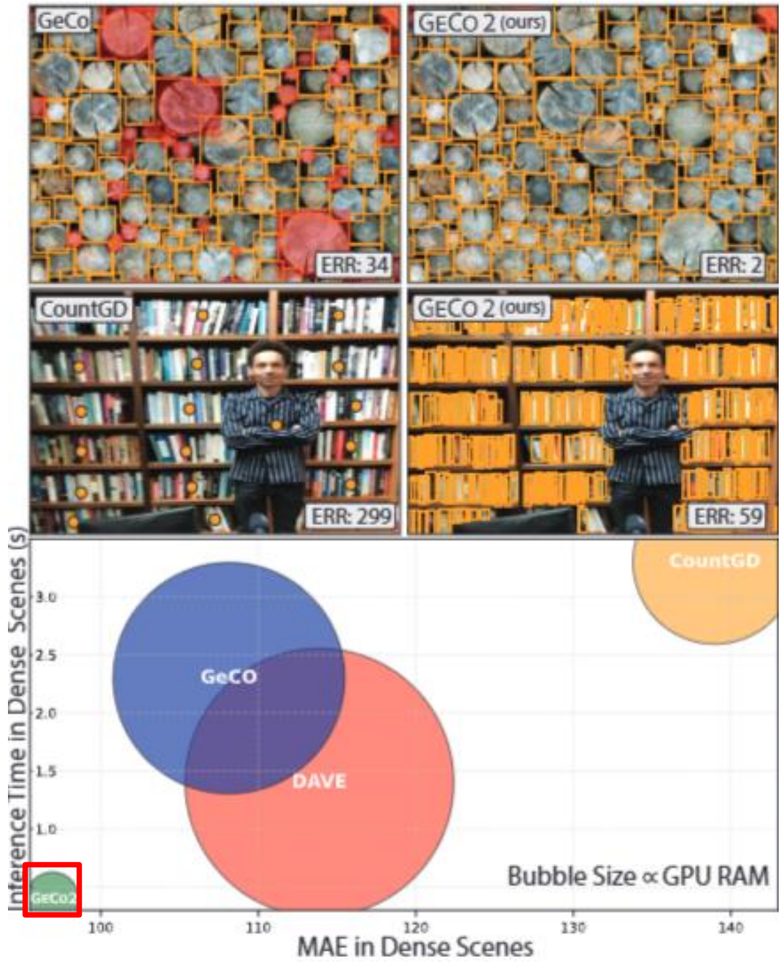
Generalized-Scale Object Counting with Gradual Query Aggregation[1)]

Introduction

- Few-shot detection-based counters : GECO2 (Generalized-Scale Counter) 제안
  - Few-shot 기반 counters 진행 방법 : 소수의 Exemplar을 사용하여 이미지 내 객체의 개수를 추정 (Main Stream)
  - 기존 진행 방법
    - 서로 다른 해상도의 backbone 특징을 병합
    - 밀집된 영역에서 작은 객체를 탐지할 수 있도록 입력 이미지를 Upsampling + 여러 조각으로 Tiling 적용
      - Tiling 적용 : Upsampling으로 발생하는 연산 및 메모리 요구 사항을 해결
      - 문제점 : 다양한 크기의 객체가 포함된 이미지나 작은 객체가 밀집된 영역에서 능력 저하
  - 신규 모델 제안 : GECO2
    - GECO2 : Multi-Scale에서 점진적으로 Query aggregation 하는 방식으로 크고 작은 객체를 모두 탐지할 수 있는 모델

Introduction

- GECO2의 Object Counter 능력 비교

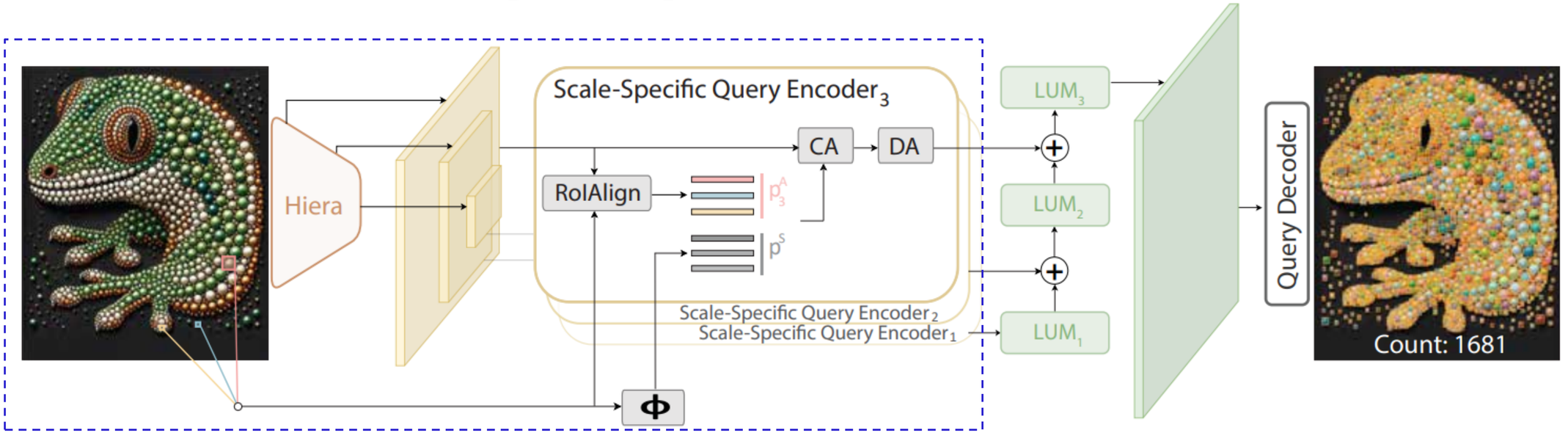


- 밀도높은 통나무 이미지에서 Counting 탐지 능력 향상
  - GECO Error 34ea → GECO2 Error 2ea
- 밀도높은 책장 이미지에서 Counting 탐지 능력 향상
  - CountGD Error 299ea → GECO2 Error 59ea

- FSC147 (GT ≥ 300ea) 기준 타모델 대비 MAE 변동 & 추론속도 우위
  - GECO2는 GECO, CountGD, DAVE 대비 밀도 높은 이미지에서 MAE 변동은 낮으면서, 추론 속도는 더 빠름

Methodology

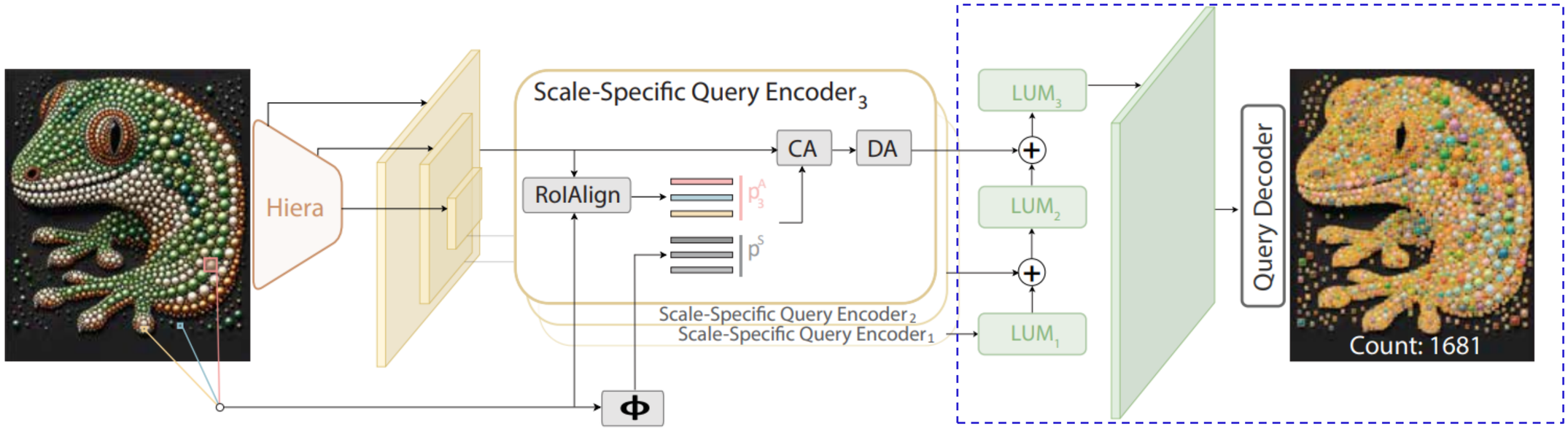
- GECO2 Structure
  - Multi-Sacle feature map 추출 (Hiera : SAM2 Vision Backbone | C_L, L ∈ {1, 2, 3})
  - Scale-Specific Query Encoder
    1. Exemplar 기준 Appearance (P^A) 및 Shape Prototype (P^S) 생성 (P_L^A : RoIAlign을 통해 추출, P^S : MLP를 통해 추출)
    2. Cross Attention (CA) : Input Feature Map C_L 과 Prototype P_L (concat(P_L^A , P^S)간의 Cross Attention ($Q'^{(i)}_L = CA(Q'^{(i-1)}_L, p_L, p_L)$)
    3. Deformable Attention (DA) : $Q''^{(j)}_L = DA(Q''^{(j)}_L)$



CA : 객체의 특징을 Query Map에 주입하여 "무엇을 찾을지" 학습
DA : 스케일별로 특화된 쿼리 맵이 완성 (연상 비용을 줄이기 위해 학습 가능한 샘플링 위치만 참고)

Methodology

- GECO2 Structure
  - Cross-Scael Query Aggregation
    - $Q = LUM_3\,(LUM_2\,(LUM_1(Q_1) + Q_2) + Q_3)$
      - LUM (Lightweight Upsampling and Fusion Module) : 서로 다른 해상도의 query map을 점진적으로 upsampling하고 결합)
        - ✓ LUM : 2 x bilinear upsamping + 3x3 conv + GeLU (Scale Query를 하나의 고해상도 query map으로 merge)
  - Dense Query Decoder : 객체 후보를 예측



Methodology

- GECO2 Structure
  - Dense Query Decoder : 객체 후보를 예측
    - Objectness score : 각 위치의 객체 존재 점수, $\mathbf{y}^{o} = \mathrm{LReLU}(\mathbf{W}^{o}\mathbf{Q})$
    - Bounding box 예측 : 각 query 위치에서 해당 객체의 bounding box를 예측, $\mathbf{y}^{\mathrm{BB}} = \sigma\big(\mathrm{MLP}(\mathbf{Q})\big)$
    - NMS : Local maximum과 threshold로 중복 후보 제거
      - 해당 위치의 objectness score가 주변 8개 이웃보다 높아야 함
      - 학습 중 정해진 threshold 보다 높아야 함 : centerness > map_max / 8 = 12.5% 초과
  - SAM2 Decoder
    - bounding box 위치 보정, 객체의 실제 픽셀 영역 추정, bounding box보다 정밀한 mask 생성, mask에 기반한 box refinement
    - SAM2가 생성한 객체 mask, $\widehat{M}_j = \mathrm{SAM2Decoder}\,(f^{I}, \hat{b}_j)$, f^{I}: backbone에서 계산된 이미지 특징, $\hat{b}_j$: GECO2가 예측한 bounding box
    - Detected Object Count > 800ea : GECO2 box 사용 (평균 객체 크기가 25pixel 이하인 경우 SAM2 mask가 불안정할 수 있음)
      - 작은 객체 탐지 : $Q_{AUX} = \mathrm{LUM}_{AUX}(Q_3)$
      - Total Loss function : $L = L' + \alpha L'_{AUX}$

Experiments

- GECO2 model implementation details

구 분	내 용
Backbone	Hiera vision backbone (SAM2에서 사용된 사전 학습된 backbone, frozen)
Input resolution	1024 x 1024
Feature map level (L)	{16, 8 , 4}
Dimension (d)	256
Exemplar Count (k)	이미지당 3개의 Exemplar Box 사용
Optimizer	AdamW (Batch Size : 8, Epochs : 200)
Learning rate	초기 10^{-4}, weight decay 5 x 10^{-5}
Hardware	NVIDIA A100 GPU 2장

Experiments

- GECO2 model results
  - Few-shot global counting (FSCD147 Dataset) : MAE 7.64

	Validation set				Test set			
Method	MAE (↓)	RMSE(↓)	AP(↑)	AP50(↑)	MAE(↓)	RMSE(↓)	AP(↑)	AP50(↑)
FamNet	23.75	69.07	-	-	22.08	99.54	-	-
CFOCNet	21.19	61.41	-	-	22.10	112.71	-	-
BMNet+	15.74	58.53	-	-	14.62	91.83	-	-
VCN	19.38	60.15	-	-	18.17	95.60	-	-
SAFEC	15.28	47.20	-	-	14.32	85.54	-	-
CounTR	13.13	49.83	-	-	11.95	91.23	-	-
LOCA	10.24	32.56	-	-	10.79	56.97	-	-
CACViT	10.63	37.95	-	-	9.13	48.96	-	-
CountGD	8.12	38.97	-	-	8.35	89.80	-	-
C-DETR	20.38	82.45	17.27	41.90	16.79	123.56	22.66	50.57
SAM-C	31.20	100.83	20.08	39.02	27.97	131.24	27.99	49.17
PSECO	15.31	68.36	32.12③	60.02	13.05	112.86	42.98③	73.33③
DAVE	9.75③	40.30③	24.20	61.08③	10.45③	74.51③	26.81	62.82
GeCo	9.52②	43.00②	33.51②	62.51②	7.91②	54.28②	43.42②	75.06②
GECo2	**9.40**①	**33.28**①	**34.08**①	**63.21**①	**7.64**①	**39.39**①	**46.81**①	**75.89**①

Experiments

- GECO2 model results
  - Few-shot global counting (FSCD147 Dataset) : MAE 7.64
    - Upscaling + tiling을 제거하면 GECO2와 타 모델간의 차이가 더 크게 발생

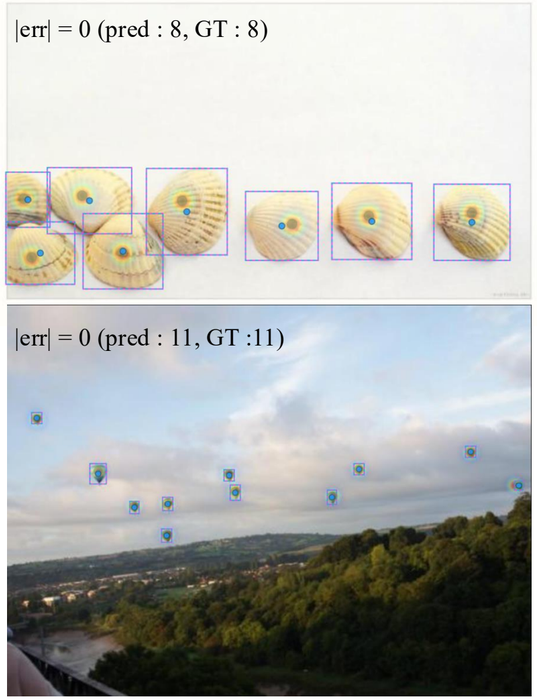
Method	MAE (↓)	RMSE (↓)	AP (↑)	AP50 (↑)
PSECO	13.05	112.86	42.98	73.33
LOCA	10.79	56.97	-	-
CountGD	9.44 ↓1.0	98.84↓9.0	-	-
DAVE	12.21↓1.8	94.93↓20.4	26.44↓0.4	61.04↓1.8
GeCo	8.88 ↓1.0	75.67↓21.4	43.15↓0.3	73.40↓1.7
GECO2	**7.64**	**39.4**	**46.81**	**75.89**

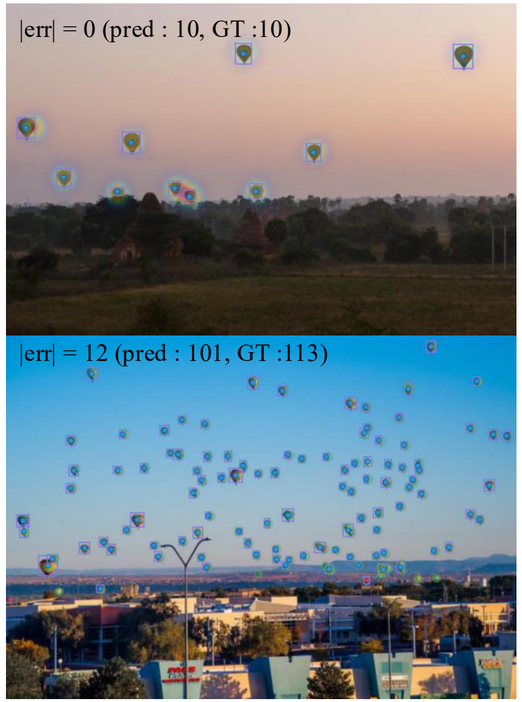
Table 4: Few-shot counters on the FSCD147 without upscaling and tiling. Performance drops are marked in red.

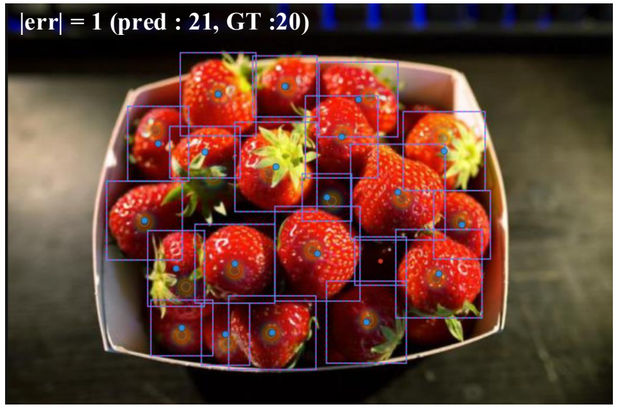
- GECO2 논문에서 제시하는 Upscaling + Tiling 문제점
  - 연산 효율성 및 자원 소모의 폭증
    - ✓ 속도 저하 : 업스케일링은 입력 데이터의 픽셀 수를 기하급수적으로 늘림
    - ✓ 메모리 부족 : Self-attention 연산 시 메모리 소모가 해상도의 제곱에 비례
  - Exemplar 추출 및 참조의 어려움
    - ✓ 잘림 현상 : Exemplar 자체가 타일 경계선에 걸려 잘릴 수 있음
    - ✓ 맥락 상실 : 이미지의 일부분(타일)만 보게 되므로, 전체 이미지의 통계적 특성이나 맥락을 반영하지 못한 채 국소적인 정보로만 비교를 수행하게 되어 오탐지가 늘어남
  - 타일 경계면의 불연속성
    - ✓ 이중 검출 : 하나의 객체가 두 타일에 걸쳐 있을 경우, 중복 카운팅 발생
    - ✓ 박스 왜곡: 경계선에 걸친 객체는 온전한 모양이 아니기 때문에, Detection-based 모델이 생성하는 Bounding Box의 좌표가 부정확해짐
  - 특징 추출의 비일관성
    - ✓ 상대적 스케일 변화: 실제 이미지에는 아주 큰 객체와 아주 작은 객체가 섞여 있을 수 있지만, 타일링은 모델에게 " 모든 객체는 이 정도 크기다 " 라고 강요하는 것과 같음
    - ✓ 고정된 시야: 업스케일링+타일링 조합은 특정 해상도에만 최적화된 추론을 유도, 이로 인해 한 이미지 내에 다양한 크기의 객체가 공존할 경우, 큰 객체는 너무 커서 특징을 놓치고 작은 객체는 타일에 맞춰 겨우 찾는 식의 불균형이 발생

Experiments

- GECO2 model results
  - Few-shot global counting (FSCD147 Dataset) : MAE 7.64
    - Best Case



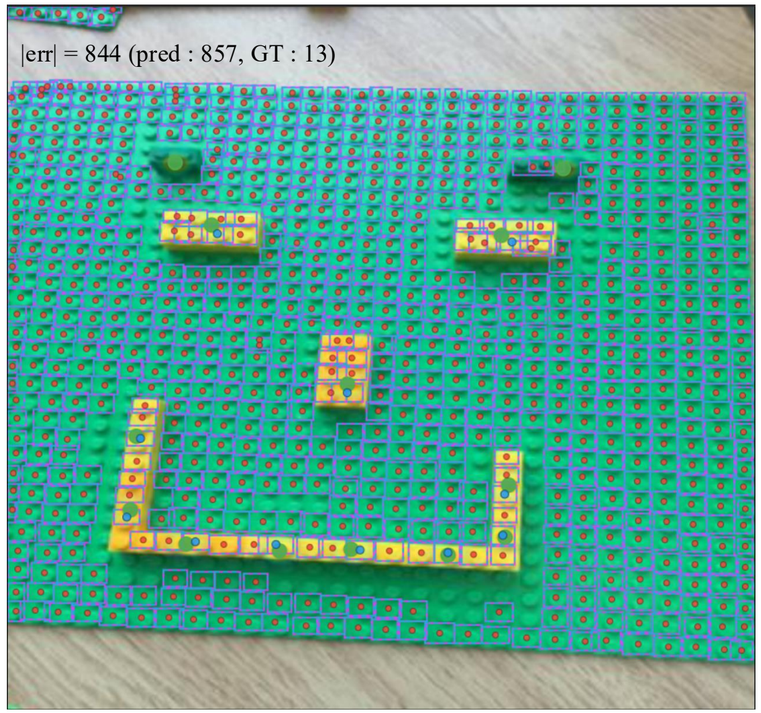


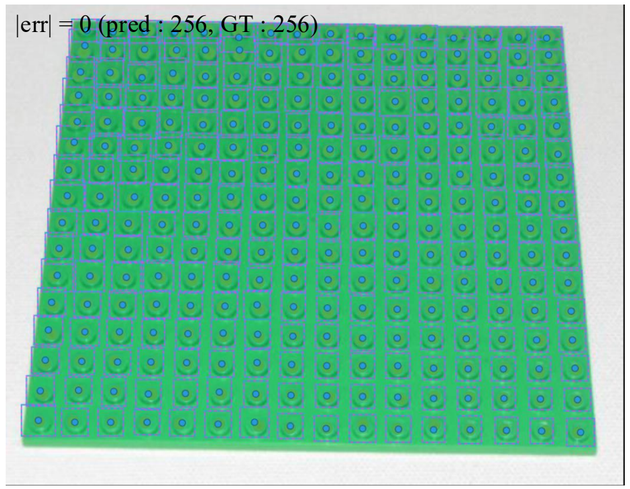


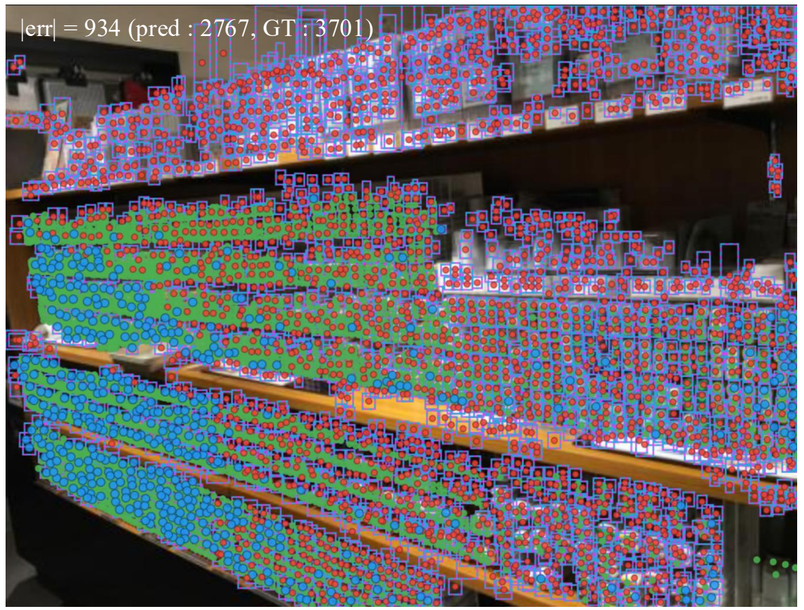
FSC147 Test dataset

Experiments

- GECO2 model results
  - Few-shot global counting (FSCD147 Dataset) : MAE 7.64
    - Worst Case







FSC147 Test dataset

Experiments

- GECO2 model ablation study
  - Few-shot global counting (FSCD147 Dataset) : MAE 7.64
    - SAM2 decode를 이용한 box refinemen를 생략하면 MAE/RMSE는 유지되지만, AP, AP50에서 감소

	Counting		Detection	
Method	MAE(↓)	RMSE(↓)	AP(↑)	AP50(↑)
GECO2	9.40	33.28	34.08	63.21
GECO2$_{FP}$	10.48	50.49	34.02	62.43
GECO2$_{\mathbf{Q_1}}$	12.71	60.29	34.05	62.10
GECO2$_{\mathbf{Q_2}}$	22.64	58.15	24.98	49.86
GECO2$_{\mathbf{Q_3}}$	19.97	55.33	24.75	47.97
GECO2$_{N_{\text{DA}=1}}$	9.45	43.19	34.03	63.05
GECO2$_{N_{\text{CA}=2}}$	11.06	45.68	33.58	62.15
GECO2$_{\alpha=0}$	10.15	44.89	33.61	61.19
GECO2$_{\alpha=0.5}$	9.39	36.74	33.47	63.02
GECO2$_{\overline{\text{REF}}}$	9.13	31.76	24.15	60.22
GECO2$_{3\Phi(\cdot)}$	10.93	53.08	34.63	63.82
GeCo+Hiera	10.99	47.99	33.75	62.84

Table 6: Ablation study on the FSCD147 val split.

Conclusion

- Exemplar Type의 Few-shot 탐지 기반 GECO2 모델 제안 (낮은 메모리 점유율)
- Upscaling, Tiling에 의존하지 않아서 계산 비용이 낮으면서 밀집된 장면에서 정확한 카운팅 능력 제공
- 기존 모델 대비 MAE를 10%, RMSE를 약 20% 감소
- AP를 10% 능가하면서도 3배 더 빠르게 실행되며 메모리 점유율은 더 낮음

1) L.Che, Z.Xue, Y.Quan, B.Liu, Z.Shi, M.Hurst, J.Feldman, R.Tang, R.Krishna, V.Pavlovic, “Counting Circuits: Mechanistic Interpretability of Visual Reasoning in Large Vision-Language Models,” arXiv preprint arXiv:2603.18523v1, 2026

Counting Circuits: Mechanistic Interpretability of Visual Reasoning in Large Vision-Language Models[2)]

Introduction

- Large Vision-Language Model (LVLM)의 추론 능력을 평가하기 위해서 Counting 이용
  - Counting은 개별 객체를 식별한 뒤 모두 합산하도록 하기 때문에 LVLM 추론 능력을 평가하기에 적절함
    - Counting은 단순한 기능이 아니라 시각적 추론의 핵심 기능 원리 : 개체 식별화 → 정보 집계
      - Object Recognition 처럼 단순히 외형을 암기하는 것보다 훨씬 복잡한 추론 과정을 포함
  - LVLM은 인간과 유사한 Counting 행동을 보임
    - 적은 수에 대해서는 정확한 성능을 보이지만 수량이 많아질수록 노이즈가 섞인 추정치를 산출
      - Subitizing : 물체의 개수를 세지 않고 즉각적으로 파악하는 뇌의 능력 (보통 1~4개의 물체는 즉각적으로 파악)
      - Noisy Estimation : 개수가 많아지면 정확도가 급격히 떨어지며 대략적인 양으로 파악
    - 이러한 인지적 불연속성은 한정된 정보 처리 자원 내에서 정밀도, 노출 시간, 주변 환경의 통계적 특성 사이의 최적점을 찾으려는 전략적 선택
      - 예를 들어 사과 3개를 볼 때는 정확히 알아야 생존에 유리하지만, 벌떼 100마리를 볼 때는 정확히 세는 것보다 '매우 많다' 라고 빠르게 판단하여 도망가는 것이 에너지 효율 측면에서 훨씬 합리적

Problem Formulation

- LVLM Counting 능력 평가 : “이미지 내의 {객체} 개수는 몇 개입니까? 숫자로만 답하세요.”
  - 객체 수는 1~10 의 범위이며 합성 데이터 셋 SyncDot, SyncPoly 와 실제 데이터 세 SyncReal 사용
  - 10개 미안의 Counting 능력에서 인간과 비교했을 때 상당히 수준이 떨어지는 것을 확인
  - 데이터 종류에 따른 역설 : SynReal > SynDot or SynPoly
    - 시각적으로 더 단순한 SynDot or SynPoly 보다 SynReal에서 더 높은 성능을 보임
    - 모델이 실제 물리적 개수를 세는 ‘추론 메커니즘’을 사용하기보다, 학습 데이터에 존재했던 통계적 상관관계에 의존하여 ‘Hallusination”식으로 답하고 있을 가능성을 시사함

Table 1: Performance comparison on synthetic benchmarks.

Model	Dataset	Acc ↑	MAE ↓	RMSE ↓	OBO ↑
Qwen2.5-VL-7B	SynDot	76.12	0.25	0.52	96.21
	SynPoly	53.74	0.94	1.75	82.27
	SynReal	73.48	0.79	2.13	91.01
Qwen3-VL-8B	SynDot	73.22	0.32	0.66	95.21
	SynPoly	70.34	0.52	1.49	94.02
	SynReal	88.79	0.15	0.47	98.88
LLaVA-1.5-7B	SynDot	24.07	4.46	6.58	39.22
	SynPoly	33.30	1.65	2.34	56.71
	SynReal	54.27	3.59	5.05	66.34

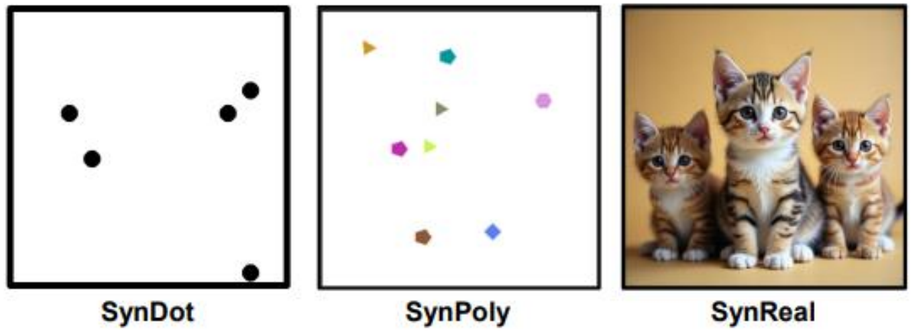


Figure 8: Data Samples of the three synthetic benchmarks.

Problem Formulation

- LVLM은 숫자를 세는가?

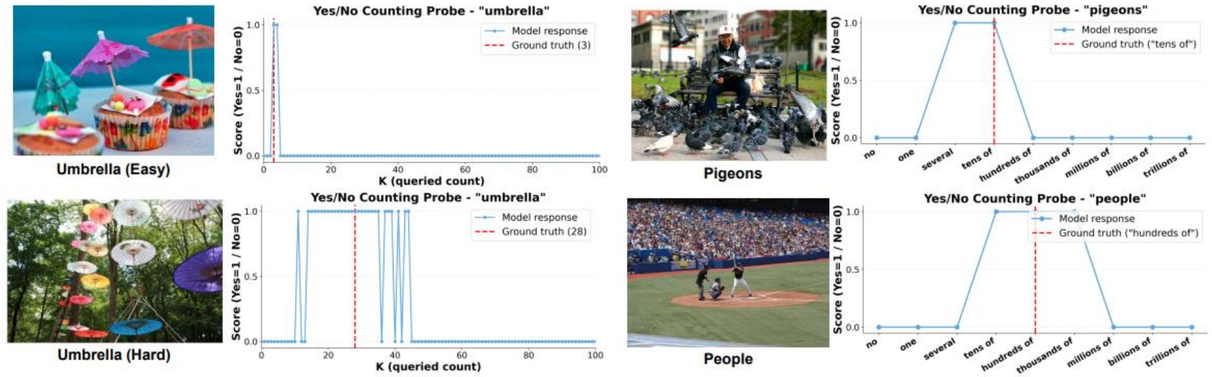


Figure 2: Counting Uncertainty Curve by Yes/No Answer of Qwen2.5 VL 7B.

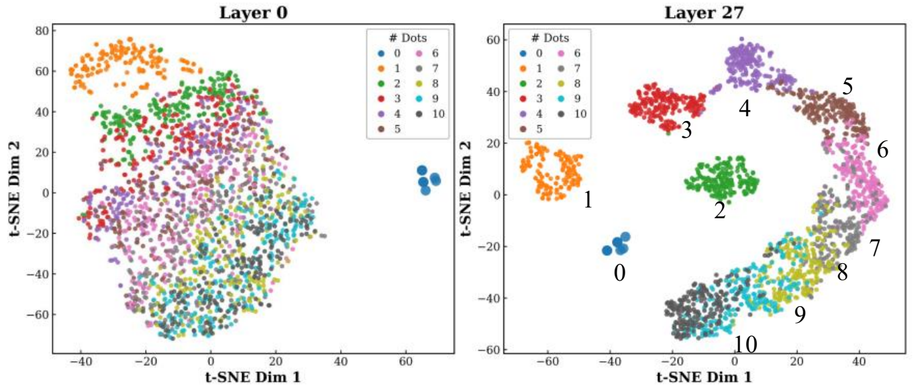


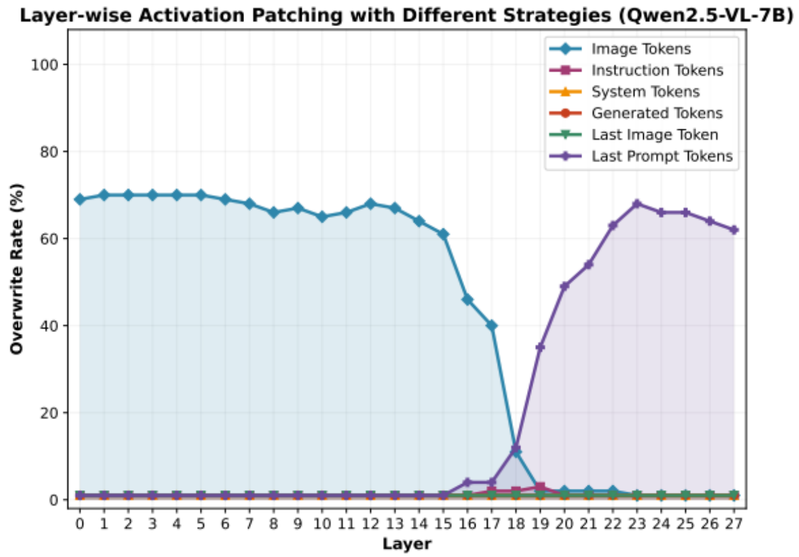
Figure 3: T-SNE visualization of model hidden states based on black dots data.

- "이미지에는 <K> 개의 <객체 이름>이 있습니다. 예 또는 아니오?"
  - K를 0~100까지 반복적으로 변경, 빽빽한 군중의 경우 K를 수십 개 또는 수백 개와 같은 정량적 설명자로 대체
  - 단순한 이미지에서는 집중되고 안정적인 YES Band
  - 복잡한 이미지에서는 결정 경계가 모호해져 Band가 확장되고 진동
  - 정밀도는 부족하지만 크기 범위를 안정적으로 구분
  - LVLM이 인간의 행동과 유사한 시각적 Subitizing 및 추정 능력 갖춤

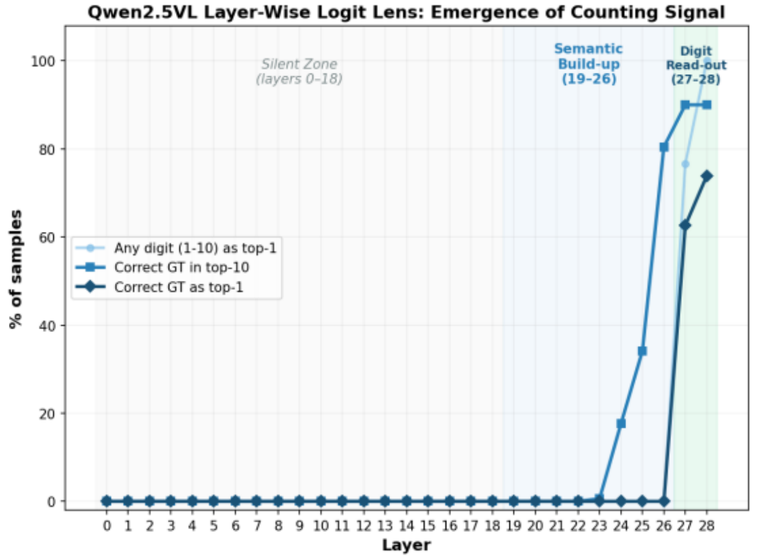
- t-SNE를 사용하여 SynDot 데이터셋의 hidden state를 2D 공간에 투영
  - 0~4개는 다른 숫자들과 분리된 뚜렷하고 고립된 클러스터 형성
  - 객체 수가 증가함에 따라 클러스터들은 점차 얽히고 심하게 중첩

Understand Layerwise Behavior of Counting

- Visual Activation Patching (VAP) & Logit Lens
  - 특정 레이어의 Hidden State를 다른 이미지의 것으로 교체했을 때 최종 출력이 얼마나 바뀌는지 검토 (a)
    - Image Tokens : 초기 레이어부터 약 15번 레이어까지 높은 영향력 유지 (Counting 정보가 Visual Features 형태로 머물러 있음)
    - Last Prompt Tokens : 15번 레이어 이후부터 영향력이 급격히 상승하여 23번 레이어 근처에서 정점을 찍음
      - 15~22번 레이어는 시각적 정보가 언어적 개념으로 변환되어 텍스트 토큰으로 넘어가는 Cross-modal Routing 구간
  - Logit Lens : 중간 레이어의 Hidden State를 모델의 최종 출력층에 통과시켜 어떤 단어가 예측되는지 확인(b)
    - LVLM의 추론이 매우 후반부 레이어에서 구체화되고 있음



(a)



(b)

Mechanistic Analysis on Attention Head Function

- Counting 작업을 지배하는 핵심 Attention Head를 확인
  - HeadLens : 수많은 Attention Head들이 각각 최종 출력에 어떤 의미적 기여를 하는지 토큰 단위로 변환하여 분석

$$\tilde{h}_i(x) = [\ \underbrace{0,\ldots,0}_{(i-1)\times d_{\text{head}}},\ h_i(x),\ \underbrace{0,\ldots,0}_{(H-i)\times d_{\text{head}}}\ ]$$

$$\hat{x} = \left(\sum_{i=1}^{H} \tilde{h}_i(x)\right) W_O = \sum_{i=1}^{H} \left(\tilde{h}_i(x) W_O\right)$$

개별 헤드 기여도 추출 : $r_i(x) = T\left(\tilde{h}_i(x) W_O\right)$

- ✓ $\tilde{h}_i$: 특정 i번째 Attention Head의 출력값 (HeadLens는 특정 헤드만 남기고 나머지는 0으로 채운 벡터를 사용)
- ✓ W_O : MHSA의 출력 투영 행렬 (개별 헤드의 결과가 모델의 전체 차원으로 어떻게 퍼져나가는지 계산)
- ✓ T(·) : 중간 레이어의 표현과 최종 레이어 표현 사이의 의미적 간극을 메워주는 역할을 하는 선형 변환($Az + b$)
- ✓ $r_i(x)$: 결과적으로 i번째 헤드가 모델의 '잔여 스트림(Residual Stream)'에 기여하는 의미적 벡터

$$r_i(x) = T\left(\tilde{h}_i(x) W_O\right) \qquad \ell_i(x) = U\, r_i(x) + c \qquad p_i(\cdot \mid x) = \text{softmax}\left(\ell_i(x)\right)$$

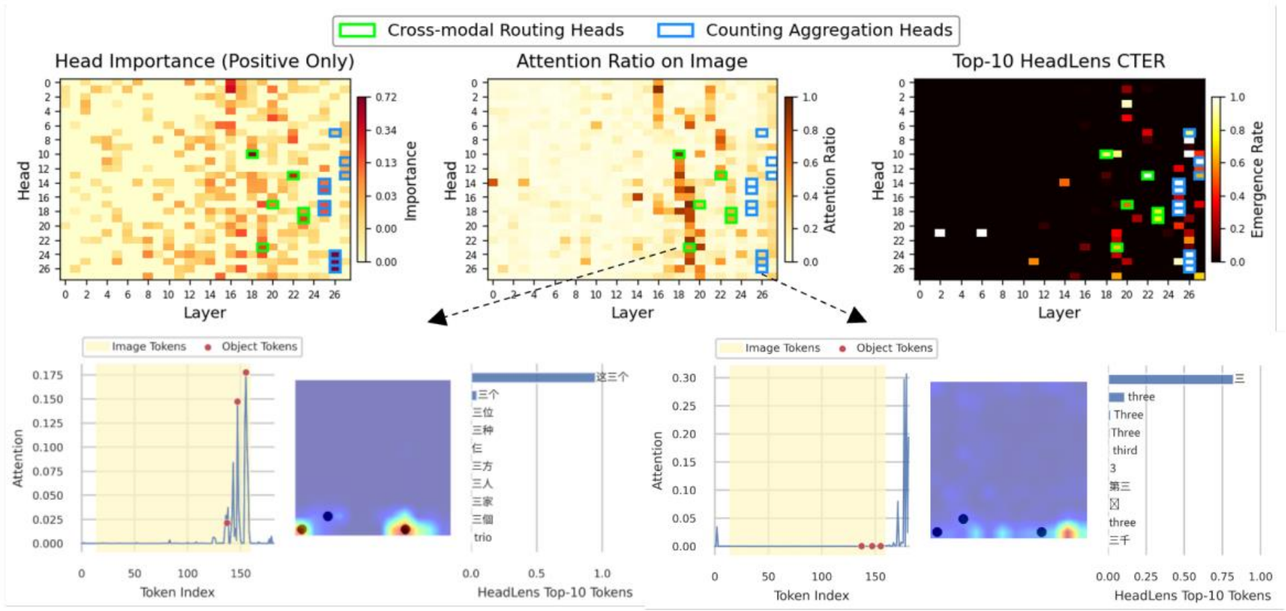
Logits 변환: $\ell_i(x) = U r_i(x) + c$

- ✓ U: 모델의 Unembedding Matrix (추상적인 벡터 공간의 값을 우리가 읽을 수 있는 단어 집합 공간으로 투영)
- ✓ $\ell_i(x)$: i번째 헤드가 투표하는 단어들의 점수(Logits) 뭉치

- HeadLens는 각 헤드의 활성화를 직접 의미론적 토큰으로 디코딩하며, HeadLens의 상위 10개 결과에서 시각적 특징 토큰 비율(Visual Grounding Score, VGS)과 계수 토큰 출현율(Counting Token Emergence Ratio, CTER)을 정의

Mechanistic Analysis on Attention Head Function

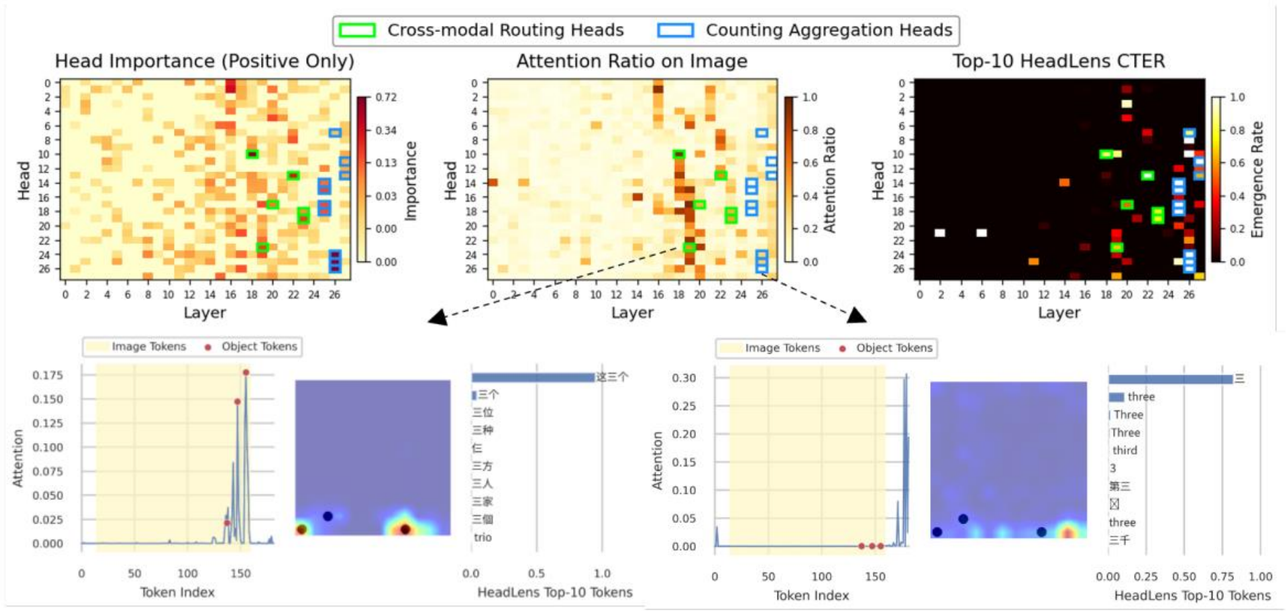
- Counting 작업을 지배하는 핵심 Attention Head를 확인
  - Cross-modal Routing Heads (초록색 상자, L19H23 예시)
    - 역할: 시각적 정보를 추상적인 숫자 개념으로 변환하여 언어 모델의 잔여 스트림으로 전달하는 가교 역할
    - 특징: 히트맵에서 이미지 어텐션 비중이 높으며, 하단 왼쪽 그래프처럼 실제 이미지 내 객체(점) 위치에 강한 어텐션을 줌
      HeadLens 결과에서도 '三个(세 개)'와 같은 숫자 의미가 나타나기 시작함
    - 비유: 요리사가 식재료(이미지)를 보고 "이건 당근 3개구나"라고 머릿속으로 판단하여 메모(텍스트 정보)를 남기는 단계



- Head Importance (좌측): VAP을 통해 특정 헤드의 활성값을 변조했을 때 최종 출력 Logit이 얼마나 변하는지 측정, 색이 진할수록 해당 헤드가 카운팅 결과에 인과적으로 중요한 영향을 줌
- Attention Ratio on Image (중앙): 해당 헤드가 전체 입력 토큰 중 Image Tokens에 얼마나 많은 어텐션을 할당하는지 보여줌
  중간 레이어(18-24층)에서 이미지에 강하게 집중하는 패턴이 관찰됨
- Top-10 HeadLens CTER (우측): HeadLens 도구를 사용하여 각 헤드의 출력을 텍스트 토큰으로 복원했을 때, 상위 10개 후보 중 정답 숫자 토큰이 포함된 비율(CTER)
  후기 레이어(26층 근처)에서 정답 숫자가 명확하게 생성됨

Mechanistic Analysis on Attention Head Function

- Counting 작업을 지배하는 핵심 Attention Head를 확인
  - Counting Aggregation Heads (파란색 상자, L26H26 예시)
    - 역할: 이전 레이어들이 처리하여 잔여 스트림에 쌓아둔 수치 정보들을 모아 최종 답변 토큰으로 집계
    - 특징: 이미지 자체에는 어텐션을 거의 주지 않지만(하단 오른쪽 그래프), 텍스트 프롬프트에 집중하며 HeadLens 디코딩 시 'three', '3' 등 매우 정확한 정답 토큰을 출력
    - 비유: 앞서 남겨진 메모들을 취합하여 최종 보고서에 "정답은 3입니다"라고 적는 최종 승인 단계



- Head Importance (좌측): VAP을 통해 특정 헤드의 활성값을 변조했을 때 최종 출력 Logit이 얼마나 변하는지 측정, 색이 진할수록 해당 헤드가 카운팅 결과에 인과적으로 중요한 영향을 줌
- Attention Ratio on Image (중앙): 해당 헤드가 전체 입력 토큰 중 Image Tokens에 얼마나 많은 어텐션을 할당하는지 보여줌
  중간 레이어(18-24층)에서 이미지에 강하게 집중하는 패턴이 관찰됨
- Top-10 HeadLens CTER (우측): HeadLens 도구를 사용하여 각 헤드의 출력을 텍스트 토큰으로 복원했을 때, 상위 10개 후보 중 정답 숫자 토큰이 포함된 비율(CTER)
  후기 레이어(26층 근처)에서 정답 숫자가 명확하게 생성됨

- **LVLM이 카운팅을 할 때 단순히 통계적 패턴을 맞추는 것이 아니라, 시각적 그라운딩 → 교차 모달 정보 변환 → 언어적 집계로 이어지는 구조화된 내부 회로를 가지고 있음**

Enhancing LVLMs' counting ability

- 단계별 Counting Mechanism 확인
  - 초기 레이어는 시각적 특징을 추출
  - 중간 레이어는 정보를 의미론적 공간으로 라우팅
  - 후기 레이어는 답을 집계

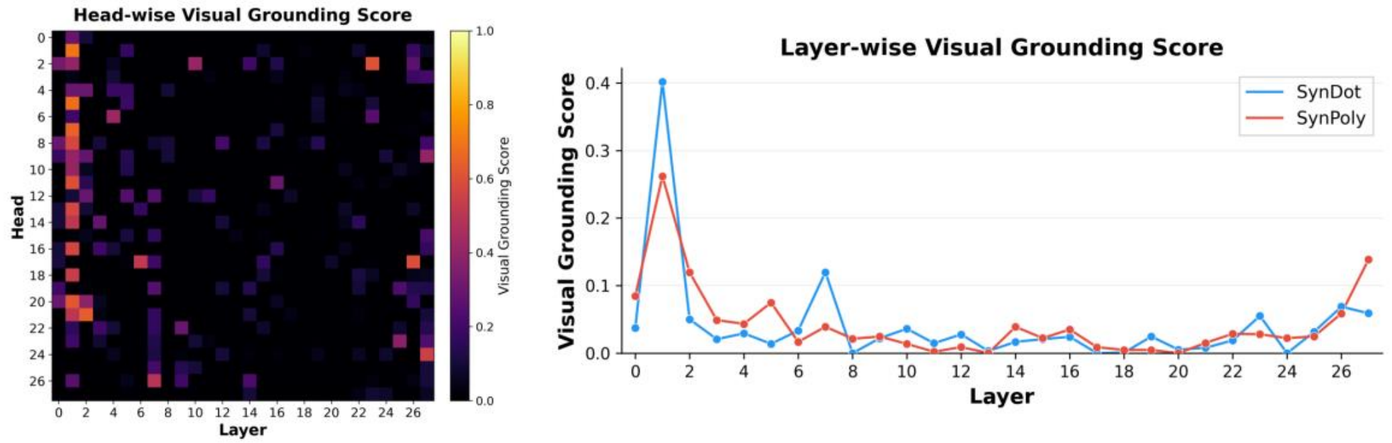


Figure 6: Visual Grounding Heads in Early Layers. Left: Head-wise heatmap based on Visual Grounding Score; Right: Layer-wise Visual Grounding Score.

Qwen2.5-VL-7B 모델 기준 SynDot, SynPoly 분석

✓ Layer1 부근에서 매우 밝은 색상 (높은 점수)이 집중되어 있음

➢ 특정 어텐션 헤드들이 이미지 토큰으로부터 직접적으로 시각적 속성을 추출하는데 특화되어 있음을 보여줌

✓ 중간층이나 후기층으로 갈수록 점수가 낮아지는데, 정보가 추상화되면서 단순한 시각 속성보다는 수치적 개념이나 집계 정보로 변환하기 때문임

✓ Layer 1에서 VGS가 최대값

➢ LVLM이 텍스트와 이미지를 결합하여 추론을 시작하기 전, 가장 초기 단계에서 객체의 경계, 색상, 형태를 파악하는 Visual Grounding 과정을 수행한다는 강력한 증거

Enhancing LVLMs' counting ability

- Adaptive Head Temperature Tuning
  - 중요한 Head들의 Attention Entropy를 줄여 관련 토큰에 대한 집중도를 높임
    - Uniform temperature scaling 대신, 식별된 대상 Head의 pre-softmax logits를 적응적으로 조절
    - Attention Entropy 감소 : 카운팅에 결정적인 역할을 하는 Core Head들을 찾아냄 → 이 헤드들이 주변의 노이즈에 휩쓸리지 않고, 카운팅에 필요한 정보(객체 위치나 수량)에만 더 날카롭게 집중하도록 만듦 (Attention Entropy를 의도적으로 낮춤)

$$A_h = \text{softmax}\left(\beta_h \frac{Q_h K_h^T}{\sqrt{d_k}}\right)$$

✓ β_h(Inverse Temperature Multiplier): logits 값을 증폭시키는 역할, $\beta_h = \alpha \times \gamma_h$
✓ α : 전체적인 엔트로피 감소 강도를 조절하는 베이스라인 계수 (1.2 사용)
✓ γ_h : 해당 헤드 h의 Importance Score (즉, 중요한 헤드일수록 더 큰 β_h를 갖게 됨)

- Object-Focused Attention Regularizer
  - 모델이 Counting 할 때 이미지의 배경이 아닌, 실제 객체가 있는 위치에 집중하로독 강제로 학습시키는 기법

$$\mathcal{L}_{\text{focus}} = \frac{1}{|\mathcal{L}||\mathcal{T}|} \sum_{l \in \mathcal{L}} \sum_{t \in \mathcal{T}} \left(-\sum_{j \in \mathcal{V}} g(j) \log\left(q_t^l(j) + \varepsilon\right) \right)$$

✓ L_{focus} : 모델이 시각적 추론을 할 때 객체의 위치를 정확히 '바라보고' 있는지 측정하는 손실 값
✓ g(j) (Ground-truth Object Prior): 우리가 모델에게 바라는 '이상적인 어텐션 분포'
  ➢ 이미지 내 객체의 중심 좌표에 Gaussian Kernel을 적용하여 만든 맵으로, 객체가 있는 위치(j)의 값은 높고 배경은 0에 가까움
✓ $q_t^l(j)$ (Predicted Attention Distribution): 레이어 l에서 토큰 t가 시각적 토큰 j를 얼마나 참조하고 있는지를 나타내는 실제 모델의 어텐션 가중치

$$q_t^l(j) = \frac{\sum_{h=1}^{H} a_h^l(t,j)}{\sum_{j' \in \mathcal{V}} \sum_{h=1}^{H} a_h^l(t,j')}, \quad j \in \mathcal{V}$$

- Loss : $\mathcal{L} = \mathcal{L}_{SFT} + \lambda \mathcal{L}_{focus}$

Experiments

- Attention Regularizer implementation details

구 분	내 용
대상 모델	Qwen2.5-VL-7B, Qwen3-VL-8B, LLaVA-1.5-7B
훈련 데이터	총 8,000개의 합성 이미지 (SynDot, SynPoly 각 4,000ea) 객체 수 1~10개, 클래스별 균등 배분
평가 데이터	OOD Counting : SynReal, PixMo-Count
최적화 함수	$\mathcal{L} = \mathcal{L}_{SFT} + \lambda \mathcal{L}_{focus}$
파라미터 미세 조정	LoRA (r = 64)
Attention Regularizer 적용 레이어	Qwen2.5-VL-7B : 2, 18~22 Layer Qwen3-VL-8B : 17~19 Layer LLaVA-1.5-7B : 0, 14 Layer
Hyperparameter	Epochs : 2, Learning Rate : 2 x 10^{-5}
훈련 환경	NVIDIA H200, BF16, AdamW
신뢰도	3개의 Random Seeds 결과의 평균값 사용

Experiments

- Attention Regularizer 적용 결과
  - OOD Counting를 일관되게 향상시킴

Backbone	Method	SynReal				PixMo-Count			
		Acc	MAE↓	RMSE↓	OBO↑	Acc	MAE↓	RMSE↓	OBO↑
Qwen2.5-VL-7B	Baseline	73.48	0.79	2.13	91.01	58.79	0.84	1.79	84.00
	Ours	**84.83**	**0.74**	**1.47**	**97.75**	**64.15**	**0.62**	**1.46**	**87.10**
Qwen3-VL-8B	Baseline	88.79	0.15	0.47	98.88	58.75	0.73	1.35	88.20
	Ours	**91.21**	**0.11**	**0.34**	**99.41**	**66.98**	**0.60**	**1.25**	**91.65**
LLaVA-1.5-7B	Baseline	54.27	3.59	5.05	66.34	31.88	1.81	3.09	59.77
	Ours	**60.11**	**1.56**	**2.08**	**91.01**	**32.64**	**1.61**	**2.59**	**62.43**

OBO (Off-By-One Accuracy) : 오차가 ±1 이내인 비율, 인간의 인지적 특성을 고려할 때 신뢰성을 측정하는 중요한 척도

Experiments

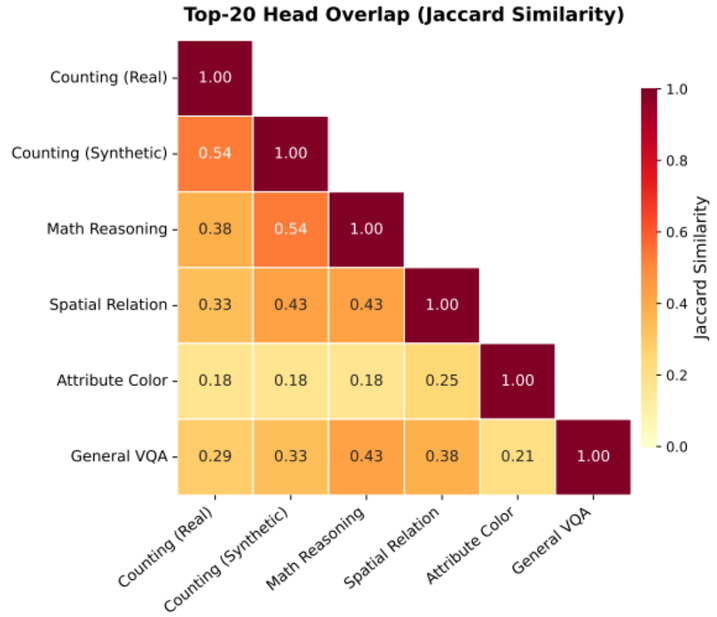
- Attention Regularizer 적용 결과
  - Counting 능력 뿐만 아니라, 일반적인 시각적 추론 능력도 향상시킴
  - Top-20 Attention Head들을 선정한 후, 두 작업 간에 이 헤드들이 얼마나 겹치는지 Jaccard Similarity를 통해 계산

$$J(HS_A, HS_B) = \frac{|HS_A \cap HS_B|}{|HS_A \cup HS_B|}$$

  - 만약 두 작업의 Jaccard Similarity가 높다면, 이는 두 작업이 모델 내에서 동일한 연산 유닛(Heads)을 공유하고 있음을 의미
  - Attribution Color는 다른 작업들과 최소한의 중첩만을 보이며, 이는 순수 지각에 의존함을 나타냄
  - Counting은 추론 중심 작업들과 중요한 헤드의 33% 이상을 공유하며, 이는 OOD 전이 가능성을 설명

Backbone	Method	MMMU	RealWorldQA	MathVista	Δ (avg.)
Qwen2.5-VL-7B	Baseline	54.89	61.96	57.30	–
	Ours	**56.33**	**64.14**	**58.30**	+1.54
Qwen3-VL-8B	Baseline	58.33	70.05	66.30	–
	Ours	**60.74**	**71.83**	**67.90**	+1.93
LLaVA-1.5-7B	Baseline	44.44	56.21	23.90	–
	Ours	**44.56**	**56.48**	**26.30**	+0.93

Table 3: Evaluation on General Capability Benchmarks.



- MMMU (Massive Multi-discipline Multimodal Understanding) : 대학 수준의 전문 지식이 필요한 고난도 시각 추론 벤치마크
- RealWorldQA : Elon Musk의 xAI에서 제안한 벤치마크로, 실제 세계의 물리적 환경에 대한 이해를 측정
- MathVista : 시각적 문맥 안에서 수학적 추론을 수행해야 하는 벤치마크

Conclusion

- Counting이 LVLM의 시각적 추론을 탐구하는 의미 있는 도구임을 증명
- Counting 관련 회로를 식별, 경량 합성 데이터 개입을 통해 Counting 기술을 향상 확인
- Counting 기술 향상이 전체 시각적 추론 능력 향상을 가져올 수 있다는 것을 증명함
- 이러한 결과를 바탕으로 앞으로 시각적 추론에 대한 통합적인 이해의 밑그림이 될 수 있는 것으로 판단함

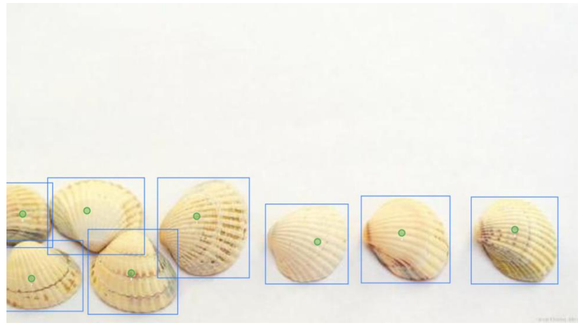
Discussion

- 모델과 인간의 시각적 구조를 고려하여 Dataset에 대해서도 구분하여 성능 검증이 필요하지 않은가?
  - 사람이 쉽게 셀 수 있는 Image와 사람이 세기 어려운 Image를 구분해서 성능을 비교할 필요가 있지 않은가?

Split	이미지 수	Class 수	캡션 수	밀집도
Train	3,659	89	3,659	49.96±85
Val	1,286	29	1,286	63.80±123.8
Test	1,190	29	1,190	66.25±146.8
Total	6,135	147	6,135	

GT Count (Test)	6-10	11-20	21-50	51-100	101-200	201-500	501-900	>900
이미지 수	60	268	413	254	139	48	6	2
비율	5.04%	22.52%	34.71%	21.34%	11.68%	4.03%	0.50%	0.17%

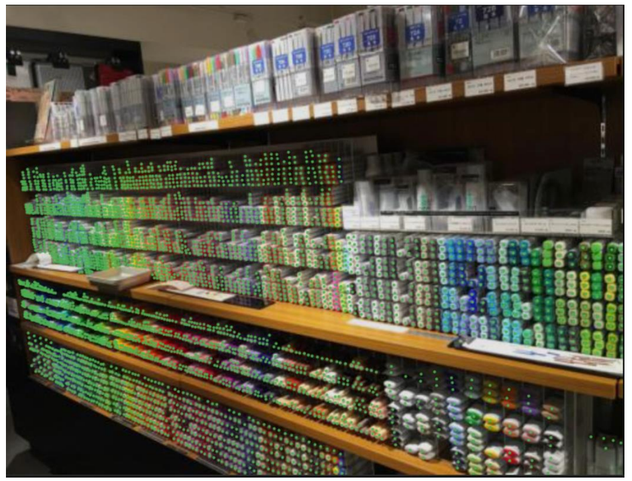
GT : 8



GT : 42

GT : 90

GT : 3701



Discussion

- 실제로 구분이 모호한 Image에 대해서 충분한 검토가 되었는가?
  - Dataset이 가지고 있는 일정한 모호함에 대해서 어디까지 이해하고 넘어가야 하는가?
  - 고성능 모델에서 일부의 Occlusion 포함/미포함에 의해 오히려 성능이 낮아지는 현상이 발생하지는 않을까?

GT : 26

GT : 28

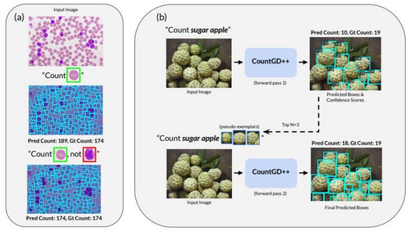
GT : 60

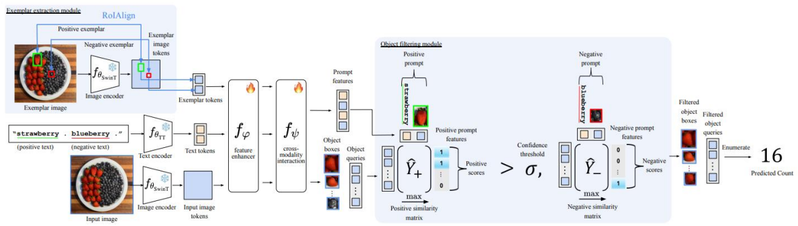
Discussion

- 특정 Image에서 발생하는 과도한 Error rate를 별도로 관리해야 하는가?
  - 과다한 Error가 발생하는 특정 Image를 맞추기 위해서 발생하는 모델 오류를 어떻게 하는 것이 좋은가?
  - 특정 Image에서 발생하는 Error rate에 의해서 전체 성능 저하가 과도하게 발생한다면 ?

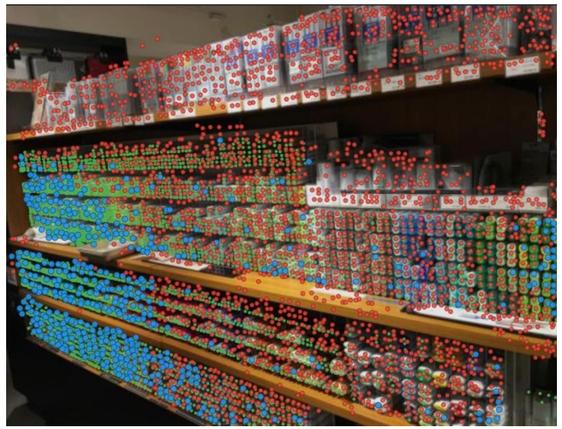
모델	추론입력	Image	MAE	RMSE	F1	AP	AP50
DAVE[1)]	3-shot bbox	Full	10.11	74.15	0.871	0.267	0.628
		excl	8.17	31.44	0.883	–	–
CountGD++[2)]	text+pseudo+synth	Full	8.28	24.58	0.874	0.389	0.715
		excl	8.28	24.60	0.906	0.421	0.769
GeCo2	GT 3-shot bbox	Full	7.71	39.56	0.866	0.463	0.735
		excl	6.92	28.86	0.892	0.499	0.783

COUNTGD++ : 객체 지정 프롬프트에 긍정 및 부정 텍스트/시각적 예시를 모두 허용하며, 추론 시 시각적 예시 주석을 자동화하는 'pseudo-exemplars' 개념을 도입하고, 외부 이미지(자연 또는 합성)에서 얻은 예시를 활용



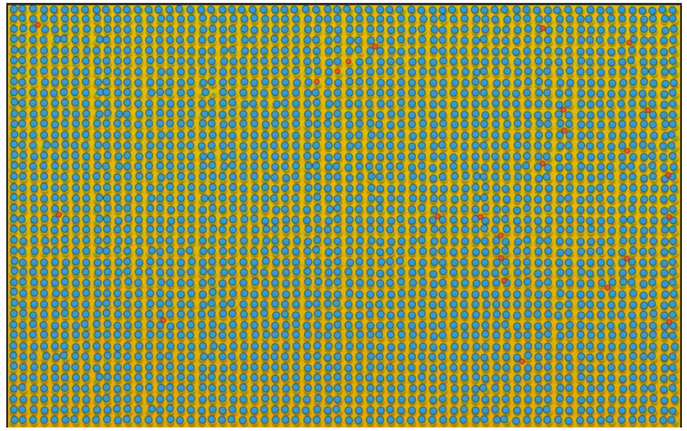


GeCo2 |err| = 934 (pred : 2767, GT : 3701)



1123.jpg markers

GeCo2 |err| = 21 (pred : 2581, GT : 2560)



7611.jpg legos

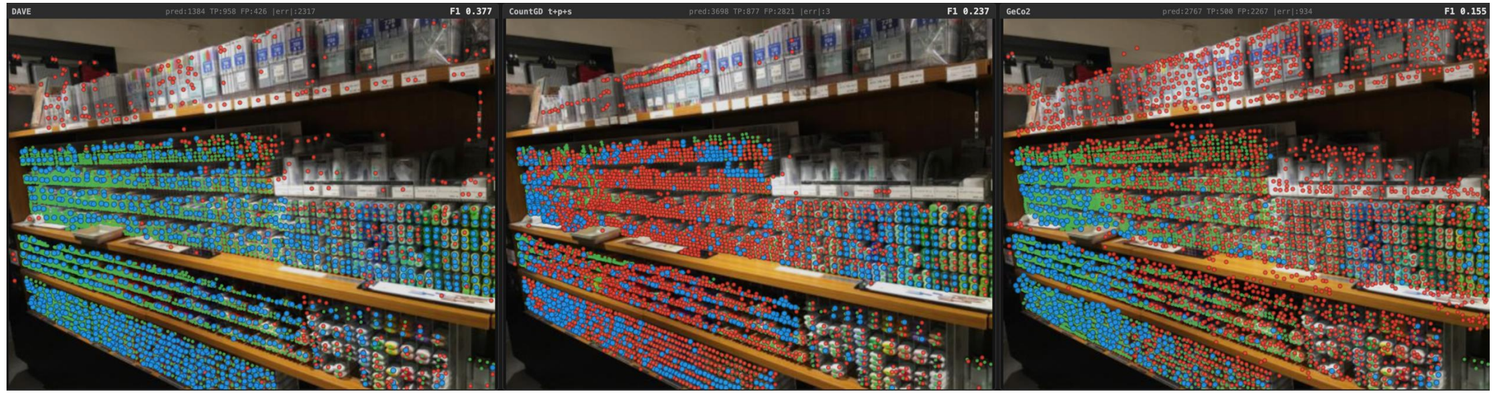
FSC147 Dataset : Test 기준 900개가 넘는 대상 2ea (1123.jpg, 7611.jpg)

Discussion

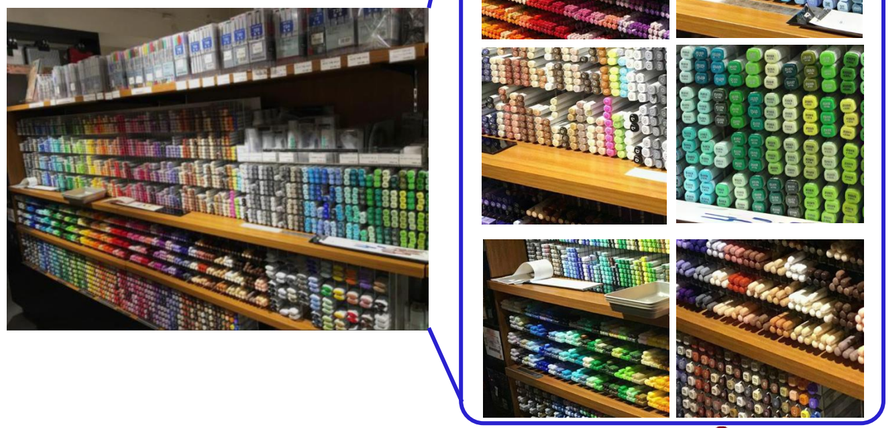
- 특정 Image에 대해서는 별도로 관리하는 것이 타당한가?
  - GroundingDino-based (backbone) model은 기본적으로 900개의 대상을 뽑을 수 있음 (query = 900ea)
    - FSC147 Test Dataset에는 900ea 넘는 2개의 이미지가 있음 (1123.jpg, 7611.jpg)
    - 1123.jpg (markers)는 이미지 해상도 자체가 낮아서 단순한 upscaling으로는 Count 진행하기 어려움
      - 일부 모델은 대상을 정확하게 Counting 하는게 아니라, 맞지 않는 대상도 Counting하여 개수를 맞추고 있다고 보여짐
  - CountGD++를 비롯한 일부 모델은 사전에 Upscaled + Crop된 Image를 모델에 넣어서 진행 (1123.jpg, 7611.jpg)

1123.jpg markers GT 3701ea

DAVE |err| = 2317 (pred : 1384 | TP:958) CountGD++ |err| = 3 (pred : 3698 | TP:877) GeCo2 |err| = 934 (pred : 2767 | TP: 500)



Upscaled + Crop (prepared in advance)



Discussion

- Seed 조건 기준으로 Variation이 많다면 타당한 결과로 볼 수 있는가?
  - 결과치의 Variation을 얼마로 설정해야 타당하다고 볼 수 있는가?
- 적정한 Epoch 설정이란 어떻게 판단해야 하는가?
  - Epoch 진행 차이에 대한 모델의 타당성을 동일한 기준에서 비교할 수 있는가?
  - Micro-burn을 적용해서 초기 epoch가 선정되지 않도록 하는 것은 괜찮은가?

구분	DAVE	CountGD++	GeCo2
학습 epoch 수	247 epochs (사전 base ep147 + fine-tune(50+50))	30 epochs	200 epochs