논문을 분석한 후 VLM(Vision-Language Model)이 사람에게는 간단해 보이는 특정 테스크에서 어려움을 겪는 이유를 아키텍처적 특징을 기반으로 추론해 드리겠습니다. 논문을 처리하는 동안 잠시 기다려 주세요.

파일을 분석한 후 결과를 공유하겠습니다. 잠시만 기다려 주세요.

논문을 분석한 결과, 현존하는 Black Boxed VLM이 사람이 보기에 간단한 테스크에서 어려움을 겪는 이유를 다음과 같이 아키텍처적인 특징에서 추론할 수 있습니다.

1. 멀티모달 정보의 비효율적인 결합

- 대부분의 VLM은 **텍스트와 이미지를 개별적으로 인코딩한 후 결합**하는 방식을 채택합니다. 하지만 이 과정에서 **이미지와 텍스트의 상관관계를 완벽하게 반영하지 못하는 문제**가 발생할 수 있습니다.
- 특히, Transformer 기반의 비전-언어 통합 모델에서는 **자연어 모델이 이미지 정보를 보조적인 요소로만 활용하는 경우**가 많아, 텍스트를 중심으로 판단하게 됩니다.
- 예를 들어, 이미지 내의 작은 차이(색상, 배경 요소, 상대적인 위치 등)를 텍스트와 효과적으로 결합하지 못하는 경우가 발생합니다.

2. 시각적 추론 능력의 제한

- 현재의 VLM은 주로 **이미지-텍스트 매칭을 학습하는 방식**을 사용하며, **논리적 추론이나 복잡한 관계 이해 능력은 제한적**입니다.
- 예를 들어, "사람이 의자에 앉아 있다"와 "의자가 사람 위에 있다" 같은 문장을 다룰 때, 일반적인 Transformer 모델은 단순한 패턴 매칭을 수행할 뿐, **물리적인 상호작용이나 공간적 관계를 고려한 논리적 판단이 어렵습니다.**
- 이러한 한계는 **공간적 인식이 중요한 테스크(예: 장면 이해, 물체 간 관계 파악)에서 성능 저하로 이어집니다.**

3. 제한된 데이터 학습과 일반화 문제

- 대부분의 VLM은 **대규모 웹 크롤링 데이터를 기반으로 학습**되며, 이 데이터는 필연적으로 편향(Bias)을 포함합니다.
- 특히, 특정한 질문이나 상황(예: **비정형적인 이미지나 보기 드문 조합**)에서 모델이 충분한 학습 경험이 없으면 **일반화에 실패**할 가능성이 큽니다.
- 이는 사람이 보기에 명확한 문제라도 VLM이 답을 제대로 도출하지 못하는 원인이 됩니다.

4. 멀티모달 어텐션의 한계

- VLM에서 **Cross-Attention**(이미지와 텍스트 간 상호작용을 조정하는 메커니즘)이 사용되지만, 이 어텐션은 **항상 최적의 방식으로 작동하지 않습니다.**
- 예를 들어, 이미지의 특정한 영역이 중요한 경우에도 모델이 이를 충분히 강조하지 못할 수 있으며, 텍스트의 특정한 부분과의 연관성을 놓치는 경우도 있습니다.
- 따라서, 단순한 객체 인식은 잘 수행하지만, **문맥적 이해나 고차원적인 비교가 필요한 테스크에서는 성능이 저하됩니다.**

5. 단순한 선형 조합을 통한 의사결정

- 많은 VLM은 최종적으로 **선형적인 분류기(Linear Classifier) 또는 회귀 모델을 통해 예측을 수행**합니다.
- 이 방식은 **비선형적인 사고 과정이나 다단계 논리 추론이 필요한 경우 약점**을 보입니다.
- 예를 들어, "X가 Y보다 크다"와 같은 상대적 개념을 이해하는 데 어려움을 겪을 수 있으며, **수학적 또는 논리적 테스크에서 VLM의 성능이 저하**될 수 있습니다.

결론

현재의 Black Boxed VLM이 간단해 보이는 문제에서 어려움을 겪는 이유는 **이미지와 텍스트 간의 비효율적인 결합, 논리적 추론 부족, 데이터 편향, 멀티모달 어텐션 한계, 선형적인 의사결정 방식** 등에서 기인합니다. 이러한 한계를 극복하기 위해서는 **더 정교한 공간적 표현 학습, 강화된 멀티모달 어텐션, 비선형적 추론 메커니즘 도입**이 필요합니다.

요약내용

다음은 논문의 각 챕터를 요약한 내용입니다.

초록

비전 언어 모델(VLMs)은 고급 이미지-텍스트 응용에서 뛰어난 성능을 보이지만, 기본적인 저수준 비전 작업에서는 인간보다 현저히 낮은 성능을 보인다. BlindTest에서 최신 VLM(GPT-4o, Gemini-1.5 Pro, Claude-3 Sonnet, Claude-3.5 Sonnet)을 테스트한 결과, 평균 정확도는 58.57%에 불과했다. 특히, 두 개의 원이 겹치는지, 선이 교차하는지, 겹치는 도형 개수를 세는 등의 작업에서 어려움을 겪었다. 가장 성능이 좋은 Sonnet-3.5조차 74.94%의 정확도로 100%에 미치지 못했다.

---

1. 서론

VLMs는 복잡한 장면 이해 및 계산 작업을 수행할 수 있지만, 기존 벤치마크들은 VLMs가 저수준 비전 작업에서도 인간 수준의 시각 인식을 갖고 있는지 평가하지 않는다. 따라서 본 논문에서는 기본적인 기하학적 도형을 활용한 BlindTest를 제안하여 VLMs의 한계를 분석했다.

---

2. 비전 언어 모델

최신 VLM 네 가지(GPT-4o, Gemini-1.5 Pro, Claude-3 Sonnet, Claude-3.5 Sonnet)를 선정하여 실험을 수행했다. 기존 VLM 벤치마크는 대학 수준의 주제, 차트, 문서, 비디오 등 고급 시각적 이해를 평가하지만, BlindTest는 단순한 기하학적 도형을 대상으로 VLMs의 기본적인 시각적 인식 능력을 검증하는 새로운 접근법이다.

---

3. BlindTest의 7가지 작업 벤치마크

BlindTest는 인간의 시력 검사를 참고하여, VLMs가 단순한 기하학적 도형을 얼마나 정확하게 인식하는지 평가하는 7가지 작업을 포함한다.

1. **선 교차점 세기** - 두 개의 선이 몇 번 교차하는지 판별.
2. **두 원의 관계 인식** - 두 개의 원이 겹치는지, 접촉하는지 판별.
3. **원으로 둘러싸인 글자 인식** - 특정 문자가 원으로 강조되었을 때 이를 정확히 판별.
4. **겹치는 도형 개수 세기** - 올림픽 로고처럼 겹치는 원과 오각형 개수 세기.
5. **중첩된 정사각형 개수 세기** - 안쪽에 겹친 사각형 개수 세기.
6. **그리드의 행과 열 개수 세기** - 빈 표와 텍스트가 포함된 표에서 행과 열 개수를 세기.
7. **지하철 경로 세기** - 두 역 사이의 단일 색상 경로 개수 세기.

---

4. 결과

1. **선 교차점 세기**: 평균 정확도 56.84%. 모델들은 두 선의 교차점을 신뢰성 있게 파악하지 못함.

2. **두 원의 관계 인식**: 평균 정확도 85.41%. 원이 가까이 있을 때 특히 오답이 많음.
3. **원으로 둘러싸인 글자 인식**: 평균 정확도 81.39%. 인접한 글자에 의해 혼동됨.
4. **겹치는 도형 개수 세기**: 원과 오각형의 경우 평균 51.35% 및 30.99%로 성능 저조. Sonnet-3.5가 가장 좋은 성능을 보였지만, 올림픽 로고의 5개 원을 편향적으로 예측하는 경향이 있음.
5. **중첩된 정사각형 개수 세기**: 평균 정확도 73.29%. Sonnet-3.5가 92.08%로 가장 우수.
6. **그리드 행과 열 개수 세기**: 빈 표에서는 34.37%, 텍스트 포함 표에서는 60.33%의 평균 정확도. VLMs는 행보다 열을 더 정확히 셈.
7. **지하철 경로 세기**: 평균 정확도 42.06%. 경로 수가 증가할수록 성능 저하.

---

5. 관련 연구

기존 VLM 벤치마크는 주로 고급 시각적 이해를 평가하는 반면, BlindTest는 단순한 기하학적 도형을 통한 저수준 시각적 인식 능력을 테스트한다. 기존 연구들은 복잡한 추론 및 자연 이미지 이해에 초점을 맞추었지만, 본 연구는 VLMs가 기본적인 시각적 구별조차 어려워한다는 점을 밝힘.

---

6. 논의 및 결론

BlindTest는 VLMs가 선, 원, 사각형과 같은 기본적인 기하학적 도형을 "볼" 수 있는지를 평가하는 새로운 벤치마크이다. 연구 결과, 최신 VLMs조차 인간이 직관적으로 해결할 수 있는 단순한 작업에서 낮은 성능을 보였다. 이는 현재 VLMs가 후반 융합(late fusion) 방식을 사용하여 시각적 정보를 처리하기 때문일 수 있으며, 향후 연구에서는 초기 융합(early fusion) 방식을 통한 개선이 필요할 수 있다. BlindTest에서 높은 성능을 보이는 VLM을 개발하는 것은 중요한 연구 방향이 될 것이다.

전체 내용

비전 언어 모델은 맹목적이다 😎

Pooyan Rahmanzadehgervi[1]★
pooyan.rmz@gmail.com

Logan Bolton[1]★
logan.bolton@auburn.edu

Mohammad Reza Taesiri[2]★
mtaesiri@gmail.com

Anh Totti Nguyen[1]
anh.ng80@gmail.com

[1] 오번 대학교, AL, USA
[2] 앨버타 대학교, 캐나다

초록. 비전 기능을 갖춘 대형 언어 모델(VLMs), 예를 들어 GPT-4o와 Gemini-1.5 Pro는 다양한 이미지-텍스트 응용 프로그램을 가능하게 하고 여러 비전 이해 벤치마크에서 높은 점수를 기록하고 있지만, 인간에게는 쉬운 저수준 비전 작업에서 놀랍게도 여전히 어려움을 겪고 있음을 발견했습니다. 구체적으로, BlindTest에서, 두 개의 원이 겹치는지 식별하기, 두 개의 선이 교차하는지 여부, 단어에서 어떤 글자가 원으로 둘러싸여 있는지, 올림픽 로고와 같은 그림에서 원의 개수를 세는 등 7개의 매우 간단한 작업을 수행할 때, 네 가지 최신 VLM은 평균 58.57%의 정확도만을 보였습니다. Sonnet-3.5는 74.94%의 정확도로 가장 좋은 성능을 보였지만, 이는 여전히 인간이 기대하는 100%의 정확도와는 거리가 멉니다. 다양한 이미지 해상도와 선 두

께에 걸쳐, VLM들은 겹치거나 가까운 기하학적 원시 도형을 인식하는 데 필요한 정확한 공간 정보가 요구되는 작업에서 일관되게 어려움을 겪고 있습니다. 코드와 데이터는 vlmsareblind.github.io에서 사용할 수 있습니다.

키워드: 비전 언어 모델 · 벤치마크 · 기하학적 원시 도형

1 서론

지난 8개월 동안, VLMs의 출현은 GPT-4V(ision)에서 시작하여 수많은, 전례 없는 이미지-텍스트 처리 응용 프로그램을 가능하게 했습니다. VLMs는 장면에서 객체를 정확히 식별하고, 이러한 탐지된 객체를 기반으로 복잡한 작업을 수행할 수 있습니다. 예를 들어, 장면 이미지와 메뉴 이미지를 통해 테이블 위 맥주의 비용을 계산하는 등의 작업을 수행합니다. 흥미롭게도, VLMs는 이미지에서 비정상적인 활동을 설명하는 것이 표준적인 건전성 검사로 빠르게 발전하고 있습니다. 기존의 VLM 벤치마크는 다양한 주제를 다루지만, 특정한 제한점을 정확히 짚어내지 않고 전반적인 인간 대 LLM의 차이를 측정합니다. 예를 들어, 많은 질문에서 입력 이미지는...

⋆ 모든 저자는 실험 수행, 결과 분석 및 논문 작성에 기여했습니다.

예를 들어, 42.9%의 MMMU [47]는 필요하지 않습니다 [12]. 즉, 많은 답변은 (1) 텍스트 질문과 선택지만으로 추론될 수 있으며 [12, 16]; (2) VLMs가 인터넷 규모의 학습에서 암기한 것입니다 [12]. 요약하면, 고수준 질문 응답에서 VLM의 우수성을 강조하면서, 현재 벤치마크는 중요한 질문을 간과합니다: VLMs는 인간처럼 이미지를 인식할 수 있는가?

이 논문에서는 VLMs가 저수준 비전 작업에서 "시력 검사" [11]를 통해 인간에게 주어진 것처럼 이미지를 볼 수 있는 능력을 테스트합니다. 우리는 4개의 최신 VLMs: GPT-4o [4], Gemini-1.5 Pro [36], Claude-3 Sonnet [10], Claude-3.5 Sonnet [6]을 7개의 매우 간단한 비전 작업(예: 선, 원)에서 테스트하며, 이는 최소한의 세계 지식을 요구합니다. 주요 발견사항은 다음과 같습니다:

1. 차트 및 다이어그램 벤치마크에서 우수한 성능에도 불구하고, VLMs는 두 선(또는 두 원)이 교차하는지를 신뢰성 있게 판별하지 못합니다, 특히 가까이 있을 때. 선 차트에서 두 개의 2-세그먼트 부분 선형 함수의 0, 1 또는 2 교차점 탐지 정확도는 약 41%에서 76%입니다 (§4.1). 두 원 작업의 경우 VLMs는 더 나은 성능을 보이지만(약 73-93% 정확도), 여전히 기대치인 100%에는 미치지 못합니다 (§4.2).

2. VLMs는 원(⭕)과 단어(Subdermatoglyphic)를 각각 완벽하게 인식할 수 있습니다. 그러나, 원이 단어에 겹쳐지면(Subder⭕ matoglyphic), 어떤 글자가 원으로 둘러싸여 있는지를 식별하는 데 어려움을 겪습니다 (§4.3).

3. VLMs는 떨어져 있는 형태, 예를 들어, 원(⭕)을 정확하게 셀 수 있습니다. 그러나, 모든 VLMs는 교차하는 원(올림픽 로고처럼)을 세는 데 어려움을 겪으며, 일반적으로 겹치거나 중첩된 원시 형태(⭕ , ■, ◇)를 어려워합니다 (§4.4).

4. 격자에 정사각형을 타일링할 때, VLMs가 그리드에서 행이나 열의 개수를 세는 데 실패하는 것을 발견했습니다, 빈 칸이든 텍스트가 포함된 칸이든 (§4.5). 이는 90% 이상의 정확도를 보이는 DocVQA [30]에서의 VLM의 높은 성능과 현저한 대조를 이룹니다 [4, 36].

5. 단순화된 지하철 지도에서 2~8개의 경로와 총 4개의 역만을 갖는 색칠된 경로를 추적하는 작업입니다. VLMs는 종종 두 역 사이의 경로를 세는 데 실패하며, 이는 약 23%에서 56%의 정확도를 보입니다 (§4.6).

6. GPT-4o는 7개의 기존 복잡한 VLM 벤치마크 [4, 36]에서는 Gemini-1.5 Pro보다 우수하지만, BlindTest에서는 더 나쁜 성능을 보입니다. 7개의 작업 전반에 걸쳐 VLMs는 평균 58.57%의 정확도를 보이며, Sonnet-3.5가 가장 좋은 성능(74.94%의 정확도)을 보이지만, 여전히 인간이 기대하는 100%의 정확도에는 한참 모자랍니다 (표 1 참조). 요약하면, BlindTest는 이전 벤치마크에서 측정되지 않은 VLM의 몇 가지 한계를 드러냅니다.

2 비전 언어 모델

우리의 목표는 상위 최신 VLMs가 서로 겹치는 기하학적 원시 도형으로 구성된 간단한 이미지를 어떻게 인식하는지를 연구하는 것입니다. 우리는 다음의 네 가지 모델을 선택했습니다: GPT-4o (🌐), Gemini-1.5 Pro (✦ Gemini-1.5), Claude-3 Sonnet (⛰️ Sonnet-3), Claude-3.5 Sonnet (🎨 Sonnet-3.5)로, 최근 7개의 멀티모달 비전 벤치마크에서 최고 순위를 기록했습니다 [4, 36]. 이들 벤치마크는 다양한 학문 분야, 대학 수준의 주제는 MMMU [47], AI2D [18]의 과학 다이어그램, MathVista [26]의 수학, ChartQA [29]의 차트, DocVQA [30]의 문서, ActivityNet-QA [46] 및 EgoSchema [27]의 비디오를 포함합니다. 초기에는 Claude 3 Opus [7]로 실험을 수행했으나, BlindTest에서 더 정확하게 수행하고 비용이 5배 적게 드는 Sonnet-3.5로 교체했습니다.

우리의 벤치마크에서는 일부 채팅 인터페이스가 API 버전보다 성능이 떨어짐을 발견했습니다(예: gemini.google.com의 시스템은 aistudio.google.com의 Gemini-1.5 Pro보다 성능이 떨어짐). 이는 추가 미세 조정 [34] 또는 VLMs를 회사의 정책에 맞추려는 특정 시스템 프롬프트 [3] 때문일 수 있습니다. 마찬가지로, GPT-4o와 Claude 3 모델이 perplexity.ai에서 원래 API 모델보다 성능이 떨어지는 것을 발견했습니다.

접근 방법 최상의 VLMs를 테스트하기 위해, 우리는 OpenAI, Google, Anthropic에서 사용 가능한 API를 통해 네 가지 모델에 접근합니다.

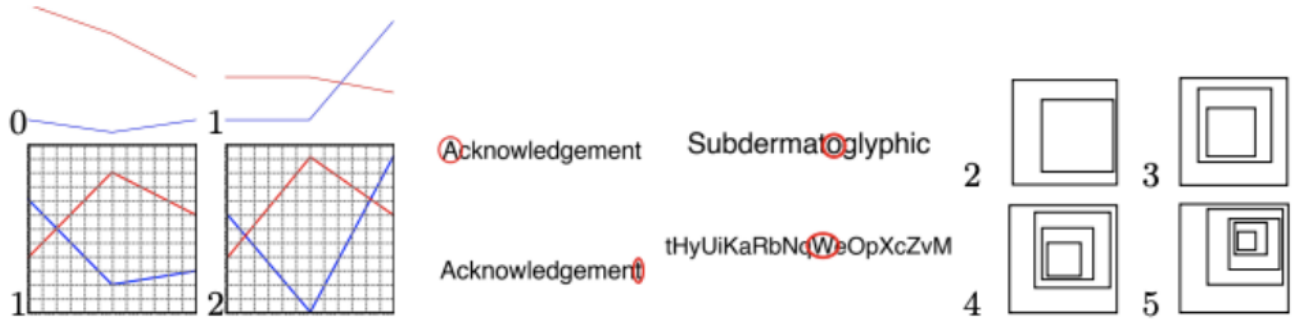
3 BlindTest의 7가지 작업 벤치마크

시력 검사 인간의 시력 검사 테스트 [11]처럼, 우리는 일반적인 기하학적 원시 도형으로 구성된 7개의 매우 간단하지만 새로운 작업 세트를 설계했습니다. 우리는 인간의 시력 검사를 위해 설계된 기존 테스트를 사용하지 않았습니다. 첫째, 인터넷에 존재하는 질문을 사용하지 않음으로써 비전 기능의 과장된 측정을 피합니다 [12, 16, 45]. 둘째, 우리의 사전 실험은 VLMs가 이미 인간의 시력 검사에서 매우 잘 수행함을 보여줍니다. 이는 일반적으로 단일, 독립된 기호(예: Snellen 차트 [11], 튜블링 E [11], 대비 민감도 차트 [8, 28])를 포함합니다.

동기 BlindTest 벤치마크는 VLMs가 가까이 있거나, 겹치거나, 교차할 때 알려진 기하학적 원시 도형을 식별하는 능력을 테스트합니다. 우리는 VLMs가 "지연 융합" [24, 39], 즉 텍스트 질문을 고려하지 않고 시각적 표현을 먼저 추출한 다음, 이를 대형 언어 모델(LLM)에 공급하는 데 주로 의존하기 때문에 어려움을 겪을 것이라고 가정합니다. 따라서 BlindTest의 기하학적 원시 도형은 잘 알려져 있지만, 흰 캔버스에 대한 정확한 공간 정보(예: 원의 크기와 위치)는 자연어로 기술하기 어려우며, 주로 자연 이미지에 대해 훈련된 비전 인코더에 의해 포착되지 않을 수 있습니다.

통제 각 테스트 이미지에 대해, 우리는 VLMs에 두 가지 다른 의미적으로 동등한 질문을 제공합니다. 또한, 세 가지 다른 이미지 크기 (§§ 3.1, 3.2, 3.4, 3.6 및 3.7)와 두세 가지 선 두께 값 (§§ 3.1 및 3.4에서 3.7)을 사용하여 원시 도형을 렌더링합니다.

3.1 작업 1: 선 교차점 세기



(a) Count intersections (b) Which letter is being red circled? (c) Count squares

Fig. 1: Images & groundtruth labels from Task 1 (a), Task 3 (b), & Task 5 (c).

차트에 대한 질문에 대한 VLMs의 뛰어난 정확도를 고려할 때(예: Sonnet-3.5가 AI2D에서 94.7% 및 ChartQA에서

90.8% 기록 [6]), 합리적인 가정은 VLMs가 두 그래프가 교차하는지 차트에서 두 개의 2-세그먼트 부분 선형 함수 사이의 교차점 수(0, 1 또는 2)를 세도록 VLMs에게 요청하여 이 가설을 테스트합니다.

이미지 우리는 1,800개의 이미지를 생성합니다 (그림 1a). 이는 크기 (C \times C)인 이미지에 그려진 2D 선 그래프이며, 여기서 (C \in {384, 768, 1152})입니다. 각 선 그래프는 두 개의 선 세그먼트로 구성되며, 세 점의 x-좌표는 ({0, \frac{C}{2}, C})px로 고정되어 있습니다 (그림 1a 참조). y 좌표는 보이지 않는 12x12 그리드에서 무작위로 샘플링되어 두 그래프 사이에 충분한 간격이 있도록 하고, 0, 1 또는 2개의 교차점이 있습니다. 자세한 내용은 §D.1을 참조하세요.

그림 1: 작업 1 (a), 작업 3 (b), 작업 5 (c)의 이미지 및 정답 레이블.

프롬프트 우리는 각 질문을 두 가지 다른 표현으로 묻습니다:

1. 파란색과 빨간색 선이 서로 몇 번 접촉합니까? 중괄호 안에 숫자로 답하세요, 예: ({5}).
2. 파란색과 빨간색 선이 만나는 교차점을 세세요. 중괄호 안에 답을 입력하세요, 예: ({2}).

정답은 (\in {0, 1, 2})입니다 (무작위 기준 정확도: 33.33%).

3.2 작업 2: 두 원

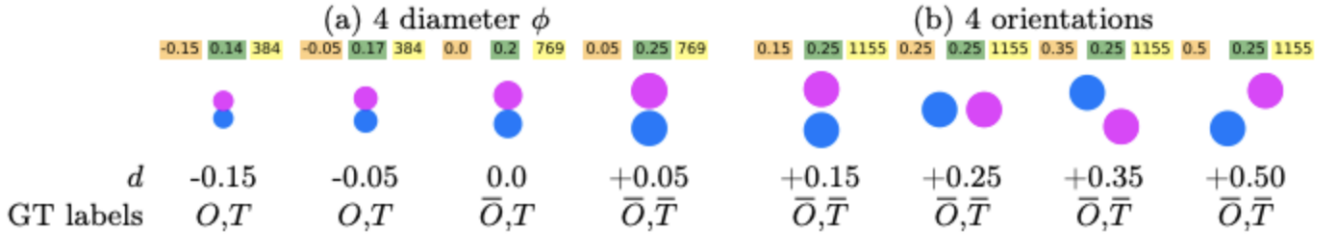


Fig. 2: For each image size and distance d, we vary diameter (a) and orientation (b). Groundtruth: O: overlapping. T: touching. $\overline{O}$: non-overlapping. $\overline{T}$: non-touching.

선 교차점을 세는 작업 (§3.1)에서, 각 이미지는 큰 흰색 캔버스에 두 개의 긴, 얇은 색깔 선을 포함합니다. 여기서는 두 상호작용하는 객체(여기서는 크기가 같은 두 개의 채워진 원 ●●)이 더 크고, 간격이 더 작아지는 보완적인 설정을 시험합니다. 이 작업은 (1) 두 원 사이의 작은 간격과 (2) 두 원이 겹치거나, 즉, 간격이 없는지를 탐지하는 VLM의 능력을 평가합니다. 우리는 원과 간격 크기를 변화시키고, 두 원이 (a) 겹치는지 또는 (b) 서로 접촉하는지를 VLMs에게 묻습니다.

이미지 크기 (C \times C)의 빈 이미지를 주어진 상태에서, 우리는 두 개의 동일한 크기의 원을 그립니다. 이들의 지름 (\phi \in {\frac{C}{4}, \frac{C}{5}, \frac{C}{7}})이며, 경계에서 경계까지의 거리 (d = \phi \times d)는 (d \in {-0.15, -0.1, -0.05, 0, 0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5})로, 세 가지 경우를 모두 포함합니다: 겹침, 접촉, 분리 (그림 2a 참조). 두 원은 네 가지 다른 방향으로 배치되어, x축과 90°, 0°, -45°, 45°의 각도를 이룹니다(그림 2b). 전체 그리드 샘플링은 이미지 크기당 224개의 이미지를 생성합니다. 우리는 이 절차를 세 가지 이미지 크기, 즉 (C = 384, 769, 1155)px에 대해 반복하여 총 (3 \times 224 = 672) 이미지를 생성합니다. 자세한 내용은 §B.1을 참조하십시오.

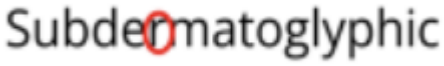
그림 2: 각 이미지 크기와 거리 (d)에 대해, 우리는 지름 (a)과 방향 (b)을 다양화합니다.
정답: O: 겹침, T: 접촉, O: 비겹침, T: 비접촉.

프롬프트

1. 두 원이 서로 접촉하나요? 예/아니오로 답하세요.
2. 두 원이 겹치나요? 예/아니오로 답하세요.

정답 두 원이 겹치고 접촉하는 경우 (d < 0.0); 비겹침이지만 접촉하는 경우 (O, T) (d = 0.0); 비겹침 및 비접촉의 경우 (O, T) (d > 0.0) (그림 2). 무작위 기준 정확도: 50%.

3.3 작업 3: 원으로 둘러싸인 글자 Subdermatoglyphic



이전 보고서와 일치하게 [37, 43, 44], 우리는 VLMs가 원시 형태(예: 빨간 원 ⭕)를 100% 정확하게 식별할 수 있으며 [37], 영어 단어(예: Subdermatoglyphic)를 단독으로 완벽하게 읽을 수 있음을 발견했습니다. 여기서, 우리는 단어의 각 글자에 빨간 원을 한 번에 하나씩 겹쳐 놓고, VLMs에게 어떤 글자가 원으로 둘러싸여 있는지를 식별하도록 요청합니다. 이 작업은 인간에게는 쉽지만, 우리의 가설은 VLM의 시야가 "흐릿"하다면 인접한 글자 사이의 작은 간격 때문에 정확히 어떤 글자가 원으로 둘러싸여 있는지를 식별하지 못할 수도 있다는 것입니다.

이미지 우리는 Acknowledgement, Subdermatoglyphic, tHyUiKaRbNqWeOpXcZvM 세 가지 문자열을 선택합니다. 이는 가변적인 너비와 높이의 문자를 포함하기 때문입니다. 게다가, 테스트한 네 가지 VLM 모두 모델에 이미지로 입력될 때 이 모든 문자를 읽을 수 있습니다. Acknowledgement는 일반적인 영어 단어이며, Subdermatoglyphic은 반복적인 글자가 없는 긴 단어입니다. 우리는 VLMs를 무작위 문자열 tHyUiKaRbNqWeOpXcZvM에서도 테스트하여 단어에 대한 친숙함에 얼마나 많은 모델 정확도가 의존하는지를 추정합니다.

각 케이스(문자열, 원으로 둘러싸인 글자 쌍)에 대해, 세 가지 빨간 원 두께 수준, 두 가지 글꼴 계열, 네 가지 값의 이미지 패딩을 사용하여 512x512 이미지를 렌더링합니다. 총 24개의 이미지를 만듭니다. 즉, 우리는 Acknowledgement(15글자), Subdermatoglyphic(17글자) 및 tHyUiKaRbNqWeOpXcZvM(20자)입니다. 각각의 글자가 타원 ⭕ 에 완전히 맞도록 보장합니다(그림 1b 참조). 자세한 내용은 §C.1을 참고하세요.

프롬프트

1. 어떤 글자가 원으로 둘러싸여 있나요?
2. 어떤 문자가 빨간 타원으로 강조되고 있나요?

정답 글자는 예측된 글자와 정확히 일치해야 합니다(대소문자 구분 없음).

3.4 작업 4: 겹치는 도형 세기 ⭕⭕⭕

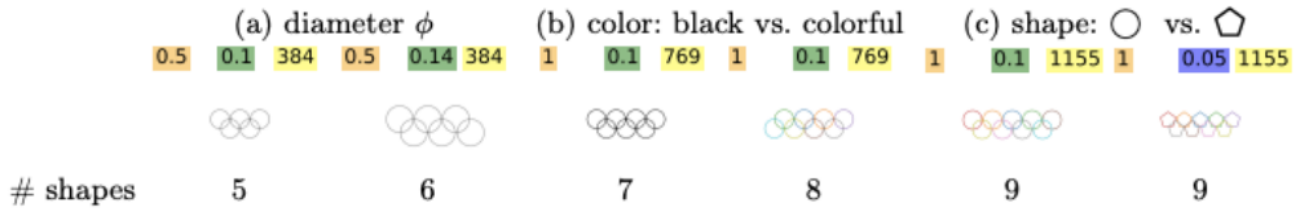


Fig. 3: Images span across three sizes and shapes span across two diameters (and two side lengths for ⬠), two color options (black vs. colored), and two line widths.

이전 연구와 일치하여 [44], 우리는 또한 VLMs가 떨어진 원(⭕⭕⭕)을 세는 능력이 있음을 발견했습니다. 그러나 여기서는 올림픽 로고처럼 교차하는 원(⭕⭕⭕)을 세는 VLMs를 테스트합니다. 이는 유아를 위한 일반적인 인지 발달 연습입니다 [1, 5]. 우리의 가설은 "흐릿한" 시야가 두 원 사이의 교차점을 명확히 보지 못해 원을 추적하고 셀 수 없을 것이라는 것입니다. 우리의 발견을 일반화하기 위해, 우리는 원 대신 오각형(◇)으로 실험을 반복합니다.

이미지 크기가 (C \times C)인 이미지에서, (C \in {384, 769, 1155})px인 경우 겹치는, 같은 크기의 원을 올림픽 로고처럼 두 줄로 배치합니다(그림 3 참조). 원의 지름 (\phi \in {\frac{C}{7}, \frac{C}{10}}). 우리는 두 가지 다른 선 두께로 원을 렌더링하여 이미지를 반복합니다. 이 절차는 3개의 해상도 × (N)의 5가지 값 × 2개의 지름 × 2개의 선 너비 × 2개의 색상 옵션 = 120개의 이미지를 생성합니다. 우리는 모든 120개 이미지를 수직으로 뒤집어 총 240개의 이미지를 만듭니다. 우리는 원 (⭕) 외에도 오각형(◇)을 반복하여 총 240x2 형태 = 480개의 이미지를 만듭니다. 오각형의 경우, 한 변의 길이 (d \in {\frac{C}{7}, \frac{C}{10}}). 자세한 내용은 §F.1을 참고하세요.

그림 3: 이미지는 세 가지 크기에 걸쳐 있으며, 도형은 두 가지 지름(및 두 가지 변 길이(◇)와 두 가지 색상 옵션(검정 대 색상), 두 가지 선 두께에 걸쳐 있습니다.

프롬프트

1. 이미지에 {도형}이 몇 개 있나요? 숫자 형식으로만 답하세요.
2. 이미지에서 {도형}을 세세요. 중괄호 안에 숫자로 답하세요. 예: ({3}).

여기서 {도형} = 원 또는 오각형입니다.

정답은 (\in {5, 6, 7, 8, 9})입니다 (무작위 기준 정확도: 20%).

3.5 작업 5: 중첩된 사각형 세기

교차하는 원을 세는 VLMs를 테스트하는 것 외에도 (§3.4), 여기서는 도형의 모서리가 교차하지 않도록 배열하는 보완 설정을 테스트합니다. 즉, 각 도형은 다른 도형 안에 완전히 중첩됩니다(그림 1c 참조). 완전성을 위해, 우리는 이 작업에서 사각형( )을 테스트합니다.

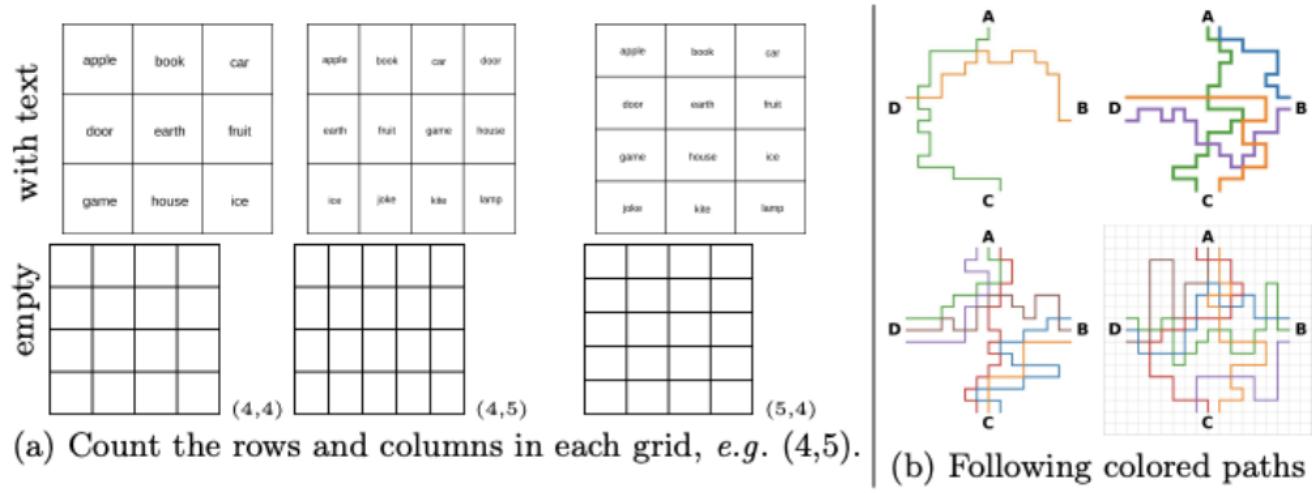
이미지 크기가 1000×1000px인 이미지에서, 우리는 (N \in {2, 3, 4, 5}) 중첩된 사각형을 가장 큰 것부터 가장 작은 것까지 하나씩 렌더링합니다. 먼저, 가장 바깥쪽 사각형은 무작위로 길이를 정한 변으로 렌더링됩니다. 그리고 각 후속 작은 사각형은 이전 사각형 내부에 무작위로 배치되며, 바깥쪽 사각형의 75% 길이의 변을 가집니다. 우리는 사각형을 선 두께가 {3, 4, 6}px인 상태로 렌더링하고, 어떤 사각형도 가장자리에 닿지 않도록 합니다. 각 선 두께에 대해, 10개의 이미지를 생성하여 (사각형이 서로 다른 무작위 위치에 있는) 3 × 10 = 30개의 이미지를 만듭니다. 모든 (N) 값에 대해 이 과정을 반복하여 4 × 30 = 120개의 이미지를 생성합니다. 자세한 내용은 §E.1을 참고하세요.

프롬프트

1. 이미지에 사각형이 몇 개 있습니까? 중괄호 안에 숫자로 답하십시오 예: ({10}).
2. 이미지에서 사각형의 총 개수를 세십시오. 중괄호 안에 숫자 형식으로만 답하십시오 예: ({3}).

정답은 (\in {2, 3, 4, 5})입니다 (무작위 기준 정확도: 25%).

3.6 작업 6: 그리드의 행과 열 세기 ■ ■ ■ ■



(a) Count the rows and columns in each grid, *e.g.* (4,5). (b) Following colored paths

Fig. 4: Empty and text-containing grids are generated with various image sizes (a). On an invisible 18×18 grid (bottom right), we randomly generate random paths from one station to another (b). All stations have an equal $N = 1$, 2 or 3 outgoing paths.

이전 작업의 결과는 VLMs가 겹치거나 (§3.4) 중첩된 (§3.5) 도형을 항상 셀 수 없다는 것을 보여줍니다. 인접한 도형은 어떨까요? 여기서 우리는 도형(특히 ■)을 그리드에 맞추고 VLMs에게 셀 수 있는 도전을 합니다. 이는 DocVQA [30]에서 90% 이상의 놀라운 성능을 보여주는 VLMs에게는 간단한 작업으로 간주됩니다. DocVQA에는 표와 관련된 많은 질문이 포함되어 있습니다. 작업을 단순화하기 위해, 우리는 모델에게 주어진 표(비어 있거나 텍스트가 포함된)에서 행과 열의 수를 세도록 요청합니다.

이미지 그리드는 (N \times N), (N \times N'), 또는 (N' \times N) 셀을 가질 수 있으며, 여기서 (N \in {3, 4, 5, 6, 7, 8, 9}) 및 (N' = N + 1)입니다. 벤치마크에서 행과 열 크기를 맞추기 위해 10×10 크기의 그리드도 포함합니다. 각 그리드는 (C \times C) 크기의 캔버스에서 두 가지 다른 선 너비로 렌더링되며, 여기서 (C \in {500, 1250, 2000})px입니다. 비어 있는 그리드 외에도, 각 셀이 단일 무작위 영어 단어를 포함하는 텍스트로 그리드를 만드는 절차를 반복합니다. 두 버전(비어 있는 것과 텍스트가 포함된 것)을 합하여 2×132 = 264개의 이미지를 만듭니다. 자세한 내용은 §G.1을 참고하세요.

프롬프트

그림 4: 비어 있는 그리드와 텍스트가 포함된 그리드는 다양한 이미지 크기로 생성됩니다 (a). 보이지 않는 18x18 그리드(오른쪽 아래)에서, 우리는 한 역에서 다른 역으로 무작위 경로를 생성합니다 (b). 모든 역은 동일하게 (N = 1, 2) 또는 3개의 출발 경로를 가집니다.

프롬프트

1. 행과 열의 수를 세고 중괄호 안에 숫자로 답하세요. 예를 들어, rows={5} columns={6}
2. 표에 행과 열이 몇 개 있습니까? 숫자 쌍(행, 열)으로만 답하세요. 예를 들어, (5,6).

정답은 행과 열의 수 모두를 포함합니다. 열과 행의 수가 모두 정확히 예측된 경우 답이 정확합니다(무작위 정확도는 1/22, 즉, 4.55%).

3.7 작업 7: 단일 색상의 경로 따라가기

VLMs가 경로를 따라갈 수 있는 것은 지도나 차트를 읽거나 [29], 그래프를 해석하거나 [19], 이미지에 있는 사용자 주석(예: 화살표)을 이해하는 데 중요합니다 [44]. 경로 따라가기 능력을 평가하기 위해, 이 작업은 모델에게 간소화된 지하철 지도에서

두 주어진 역 사이의 고유 색상 경로의 수를 세도록 요청합니다.

이미지 각 지하철 지도를 (C \times C) 크기의 이미지로 생성하며, 여기서 (C \in {512, 1024})px입니다 (그림 4b 참조). 네 개의 고정된 좌표에 네 개의 역 이름(A, B, C, D)을 씁니다: ({(\frac{C}{2}, C), (C, \frac{C}{2}), (\frac{C}{2}, 0), (0, \frac{C}{2})}). 캔버스를 보이지 않는 18x18 그리드로 나누고 각 역에서 (\frac{C}{18})px 떨어진 3개의 경로 시작점을 초기화합니다. 깊이 우선 탐색 알고리즘을 사용하여 무작위 역과 무작위 시작점에서 경로를 그리며, 유효한 이동은 북쪽, 남쪽, 동쪽 또는 서쪽으로 한 셀입니다. 각 역이 정확히 (N \in {1, 2, 3})개의 출발 경로를 갖도록 과정을 반복하여 총 180개의 지도를 만듭니다. 자세한 내용은 §H.1을 참조하십시오.

프롬프트

1. A에서 C로 가는 단일 색상 경로가 몇 개 있습니까? 중괄호 안에 숫자로 답하세요. 예: ({3}).
2. A에서 C로 가는 단일 색상의 경로를 세십시오. 중괄호 안에 숫자로 답하세요. 예를 들어, ({3}).

정답은 ({1, 2, 3})에 속합니다 (무작위 기준 정확도: 33.33%). 작업을 쉽게 만들기 위해, 0은 정답 세트에서 제외됩니다(즉, 두 주어진 역 사이에 경로가 없는 경우 모델에게 세도록 요청하지 않습니다).

---

4 결과

표 1: BlindTest에서 7가지 작업에 대한 각 모델의 정확도(%). 모든 네 가지 모델의 평균 정확도는 58.57%로, 무작위 기회보다 상당히 더 높습니다(24%). 이는 각 작업을 단일 레이블, N-방향 분류 문제로 고려하여 계산됩니다. Sonnet-3.5는 가장 좋은 성능을 보입니다(74.94% 정확도), 그러나 여전히 100%의 기대 정확도에는 미치지 못합니다.

Model	a.	b.	c.	d.	e.	f.	g.	h.	i.	Task mean
Random	33.33	50.00	5.77	20.00	20.00	25.00	4.55	33.33	24.00	
Model 1	41.61	72.67	70.18	42.50	17.50	55.83	39.58	47.89	48.47	
Model 2	66.94	92.78	92.81	87.08	19.37	80.00	39.39	41.60	65.00	
Model 3	43.41	84.52	73.34	31.66	9.79	65.00	36.17	23.24	45.89	
Sonnet-3.5	75.36	91.66	89.22	44.16	77.29	92.08	74.26	55.53	74.94	
Mean	56.84	85.41	81.39	51.35	30.99	73.29	47.35	42.06	58.57	

4.1 작업 1: VLMs는 교차점을 신뢰할 수 있게 세지 못합니다

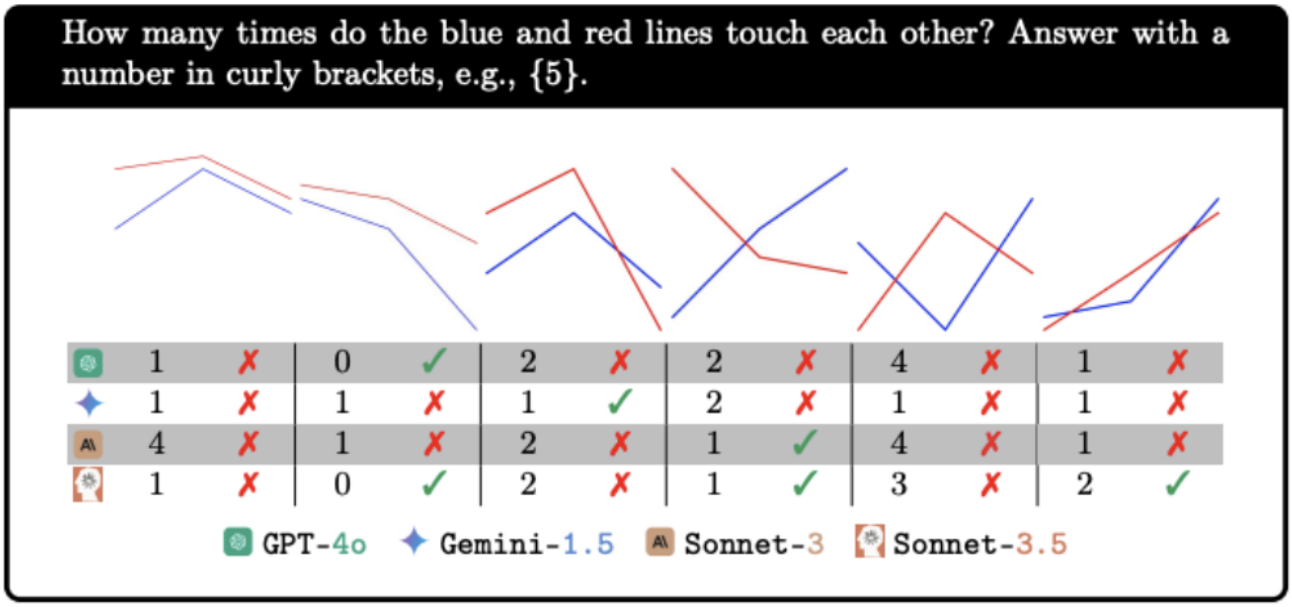


Fig. 5: VLMs cannot reliably count the intersections between the blue and red plots.

Table 2: The accuracy breakdown by line width in pixels (where C = image width), averaged over two prompts, shows that VLMs cannot reliably count the intersections between two simple 2D line plots.

Line width				
$0.005 \times C$	45.00	67.55	45.22	75.83
$0.010 \times C$	38.22	66.33	41.61	74.88
Mean	41.61	66.94	43.41	**75.36**

Fig. 6: As a line-plot image is divided into a 12×12 grid, the x-axis shows the mean **distance** (in grid cells) over 3 pairs of points of two 2-segment plots. VLMs are often more confused when two plots are closer together (left) than when they are further apart (right).

실험 우리는 모든 모델의 응답을 파싱하여 최종 답변을 추출하고 이를 정답과 비교합니다. 우리는 두 개의 프롬프트에 대한 각 모델의 평균 정확도를 보고하고, 하이퍼파라미터(예: 선 너비 및 이미지 크기)를 변경할 때 정확도가 어떻게 변하는지 분석합니다.

결과 먼저, 두 개의 프롬프트와 두 가지 선 너비에 걸쳐, 모든 VLMs의 정확도는 56.84%로, 이 쉬운 작업에 대한 기대 정확도인 100%에는 한참 미치지 못합니다(그림 5). 최고 정확도는 75.36%(Sonnet-3.5)입니다(표 2). 특히, VLMs는 두 개의 플롯 사이의 거리가 좁아질 때 더 나쁜 성능을 보이는 경향이 있습니다(그림 6). 각 선 플롯은 세 개의 핵심 점으로 구성되어 있으며, 두 플롯 간의 거리는 세 쌍의 해당 점 사이의 평균 거리로 계산됩니다. VLM 예측의 더 많은 샘플은 §D.4를 참조하세요. VLM은 세 가지 이미지 크기에서 유사하게 작동합니다(§D.2).

우리의 발견은 ChartQA에서 VLMs의 높은 정확도와 현저히 대조됩니다 [4, 36], 이는 VLMs가 선 플롯의 전체적인 경향을 인식할 수 있음을 시사하며, 예를 들어 선이 교차할 때 "확대"하여 세부 사항을 볼 수 없음을 제안합니다.

질문: 파란색과 빨간색 선이 서로 몇 번 접촉하나요? 중괄호 안에 숫자로 답하세요, 예: ({5}).

그림 5: VLMs는 파란색과 빨간색 플롯 사이의 교차점을 신뢰할 수 있게 셀 수 없습니다.

표 2: 픽셀 단위로 선 너비에 따른 정확도 분포(여기서 (C)는 이미지 너비)를 두 개의 프롬프트에 대해 평균화한 결과, VLMs는 두 개의 단순한 2D 선 플롯 사이의 교차점을 신뢰할 수 있게 셀 수 없음을 보여줍니다.

Line width	GPT-4o	Gemini-1.5	Sonnet-3	Sonnet-3.5
(0.005 \times C)	45.00	67.55	45.22	75.83
(0.010 \times C)	38.22	66.33	41.61	74.88
Mean	41.61	66.94	43.41	75.36

그림 6: 선 플롯 이미지를 12×12 그리드로 나눌 때, x축은 두 개의 2-세그먼트 플롯의 3쌍의 점에 대한 평균 거리(그리드 셀 단위)를 보여줍니다. VLMs는 플롯이 더 가까이 있을 때(왼쪽)보다 더 멀리 떨어져 있을 때(오른쪽) 더 혼란스러워하는 경향이 있습니다.

4.2 VLMs는 두 원이 겹치는지 명확히 볼 수 없습니다

VLM의 선 교차점 세기 성능이 저조한 것에 동기부여되어 (§4.1), 여기서는 큰 색칠된 원을 사용하여 두 원이 접촉(또는 겹침)하는지를 VLMs에게 명확히 물어봅니다.

실험 VLMs에게 이진 답변(예/아니오)을 출력하도록 지시하므로, 우리는 Python을 사용하여 VLMs의 응답에서 형식화된 답변을 추출하고 정답과 비교합니다.

결과 놀랍게도, 객체(●●)가 크고 사람에게 명확히 보이는 이 작업에서도, 모든 VLMs는 평균 정확도 85.41%로 완벽하게 해결할 수 없습니다(표 1b). 최고의 정확도는 모든 이미지와 두 프롬프트에 대해 92.78%(Gemini-1.5)입니다(표 3). 일반적인 경향은 두 원이 서로 가까이 있을 때, VLMs는 잘못된 성능을 보이며, 예를 들어, 추측하여 답변하는 경향이 있습니다.

질문: 두 원이 겹쳐져 있습니까? 예/아니오로 답하세요.

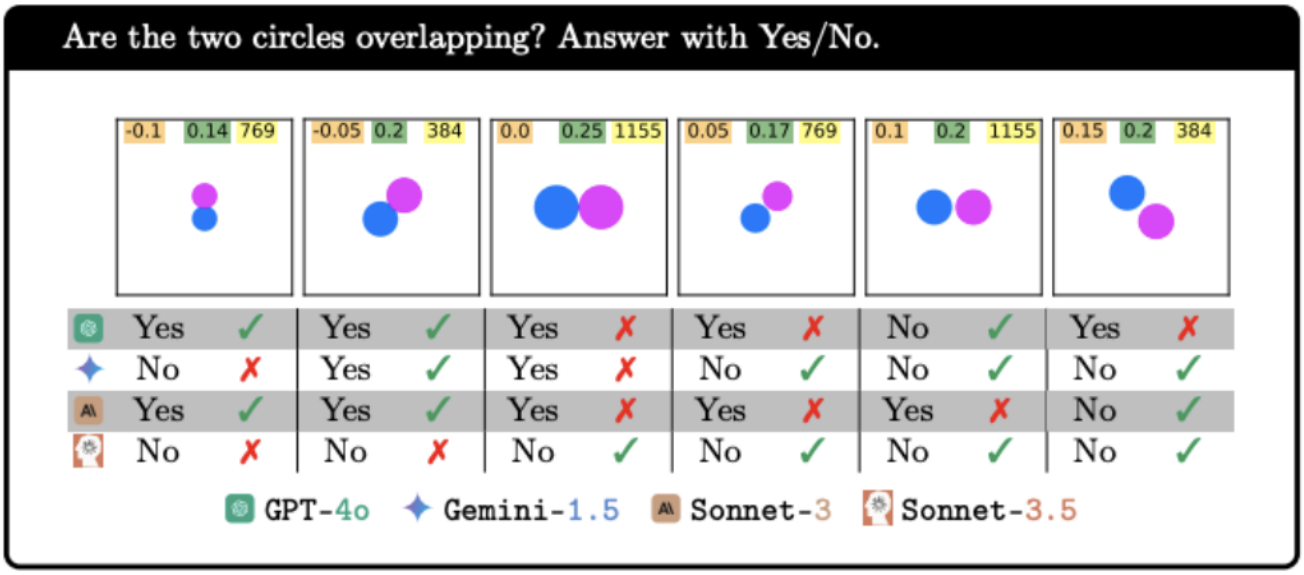


Fig. 7: VLMs consistently fail at smaller distances. However, when the gap is large and clearly visible, GPT-4o remains unreliable. Sonnet-3.5 tends to conservatively answer "No" regardless of the actual distance between the two circles.

그림 7: VLMs는 더 작은 거리에서 일관되게 실패합니다. 그러나 간격이 크고 명확하게 보일 때도, GPT-4o는 여전히 신뢰할 수 없습니다. Sonnet-3.5는 두 원 사이의 실제 거리에 상관없이 보수적으로 "아니오"라고 답하는 경향이 있습니다.

Sonnet-3.5는 종종 보수적으로 "아니오"라고 답합니다(그림 7). GPT-4o는 가장 성능이 낮으며, 두 원 사이의 거리가 반경만큼 큰 경우에도 100% 정확하지 않습니다(그림 8; (d = 0.5)). 즉, §4.1의 결과와 일치하게, VLMs는 항상 두 채워진 원 사이의 간격이나 교차점을 감지할 수 없는 것처럼 보입니다(그림 7).

한 가지 설명은 후반 융합 메커니즘 [39]으로 인해, VLMs가 질문을 보기 전에 이미지에서 시각적 특징을 추출하여 이 "맹목"을 초래한다는 것입니다. 반대로, 만약 모델이 질문이 두 원 사이의 영역에 집중하라는 것임을 먼저 안다면, 그러한 간단한 질문에 답하기 위해 정확한 시각 정보를 추출할 수 있을 것입니다.

VLMs는 세 가지 이미지 해상도에서 일관되게 작동하지만 (§B.2), 모든 모델은 특정 원 방향에서 가장 잘 수행합니다 (§B.5). VLMs의 답변 예시는 §B.7에 더 많이 있습니다.

Table 3: GPT-4o and Gemini-1.5 perform more consistently over the two different prompts ("*overlapping*" and "*touching*") than Sonnet-3 and Sonnet-3.5.

Model	Overlapping	Touching	Mean
GPT-4o	71.27	74.10	72.69
Gemini-1.5	**93.30**	92.26	**92.78**
Sonnet-3	88.09	80.95	84.52
Sonnet-3.5	88.83	**94.49**	91.66

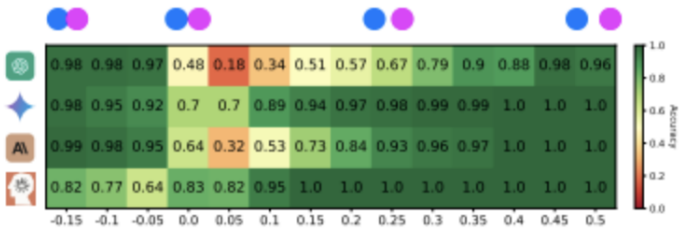


Fig. 8: VLMs perform poorly when two circles are tangent ($d = 0.0$) or close together ($d = 0.05, 0.10$). Yet, Sonnet-3.5 is better at $d \geq 0.0$. (perhaps due to its tendency to answer "No").

표 3: GPT-4o와 Gemini-1.5는 두 가지 다른 프롬프트("겹침" 및 "접촉")에서 Sonnet-3 및 Sonnet-3.5보다 더 일관되게 수행합니다.

Model	Overlapping	Touching	Mean

Model	Overlapping	Touching	Mean
GPT-4o	71.27	74.10	72.69
Gemini-1.5	93.30	92.26	92.78
Sonnet-3	88.09	80.95	84.52
Sonnet-3.5	88.83	94.49	91.66

그림 8: VLMs는 두 원이 접하는 경우((d = 0.0)) 또는 가까이 있는 경우((d = 0.05, 0.10))에 성능이 저조합니다. 그러나 Sonnet-3.5는 (d \geq 0.0)에서 더 나은 성능을 보입니다(아마도 "아니오"로 답하는 경향 때문일 수 있습니다).

4.3 VLMs는 항상 빨간 원 안의 글자를 보지 못합니다

실험 각 모델의 출력은 특정 형식을 따르므로, 우리는 정규식 매칭과 수동 검토를 사용하여 답변을 추출합니다. 예를 들어, GPT-4o의 응답은 일반적으로 예측된 글자가 따옴표(') 안에 포함된 한 줄의 설명을 포함합니다. 그러나 Gemini-1.5는 마크다운 형식을 사용하고 글자 주위에 이중 별표(**)를 배치합니다.

결과 먼저, 모든 VLMs는 빨간색 타원형이 중첩된 문자열을 정확히 철자할 수 있습니다. 그러나 흥미롭게도, 어떤 글자가 원으로 표시되고 있는지를 읽는 것은 도전 과제가 됩니다(모델 평균 정확도: 81.39%; 표 1c). 예를 들어, VLMs는 인접한 글자가 빨간색 타원형에 의해 부분적으로 가려질 때 실수를 저지르는 경향이 있습니다(예: Subderm◯atoglyphic). 더 많은 실패 사례는 그림 9에서 볼 수 있습니다.

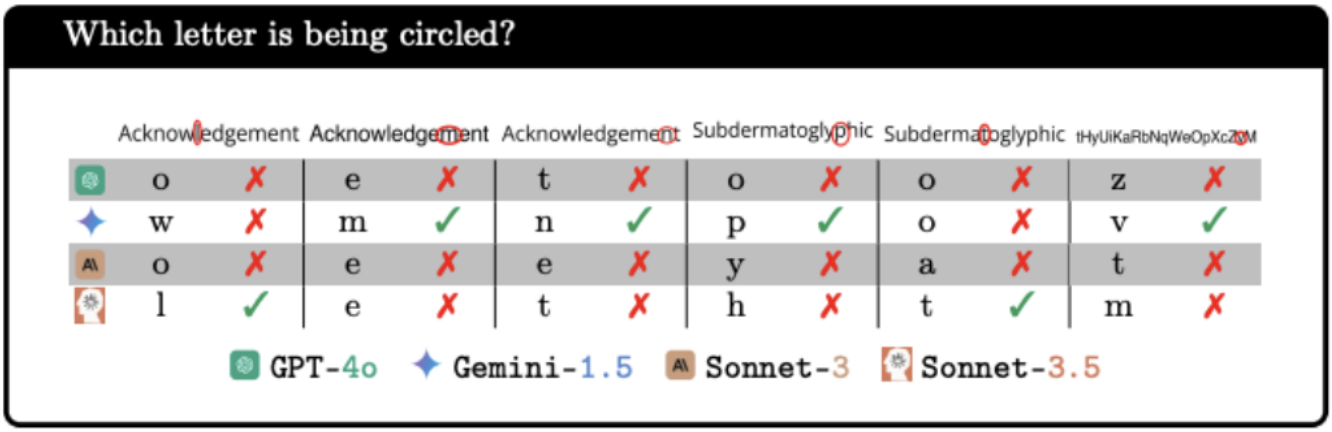


Fig. 9: Identifying the letter being circled is non-trivial for VLMs across both English words (Acknowledgement & Subdermatoglyphic) and a random string (tHyUiKaRbNqWeOpXcZvM). When making mistakes, VLMs tend to predict letters adjacent to the circled one.

그림 9: 원으로 표시되고 있는 글자를 식별하는 것은 두 영어 단어(Acknowledgement 및 Subdermatoglyphic)와 임의의 문자열(tHyUiKaRbNqWeOpXcZvM)에서 VLMs에게 비상식적입니다. 실수를 할 때, VLMs는 원으로 표시된 글자에 인접한 글자를 예측하는 경향이 있습니다.

실수를 할 때, VLMs는 종종 원으로 표시된 글자에 인접한 글자를 예측합니다(그림 10의 혼동 행렬 및 §C.2의 더 많은 결과 참조). 때때로 모델은 환각하여, 예를 들어 존재하지 않는 문자(Subdermatoglyphic에서 "9", "n", "©")를 만들어냅니다. 비록 그것이 단어를 정확히 철자할 수 있음에도 불구하고(그림 10 참조). 더 많은 실패 사례는 §§ C.3 및 C.6에 보고됩니다.

평균적으로, 모델은 임의의 문자열에 비해 두 영어 단어에서 더 나은 성능(+2에서 +7포인트)을 보이며(표 4), 이는 단어를 아는 것이 VLMs가 더 잘 추측하게 하고 약간의 정확도 향상을 제공함을 시사합니다. 또한, 그림 17에서 겹치는 두 원 사이의 영역 색상에 대한 GPT-4o와 Gemini-1.5의 예측을 참조하십시오(작업 1).

Gemini-1.5와 Sonnet-3.5는 상위 두 모델(92.81% 및 89.22%)이며, GPT-4o와 Sonnet-3보다 거의 20포인트 더 높은 성능을 보입니다.

(Table 4). VLMs perform similarly across two prompts (§C.5) and two font families (§C.4).

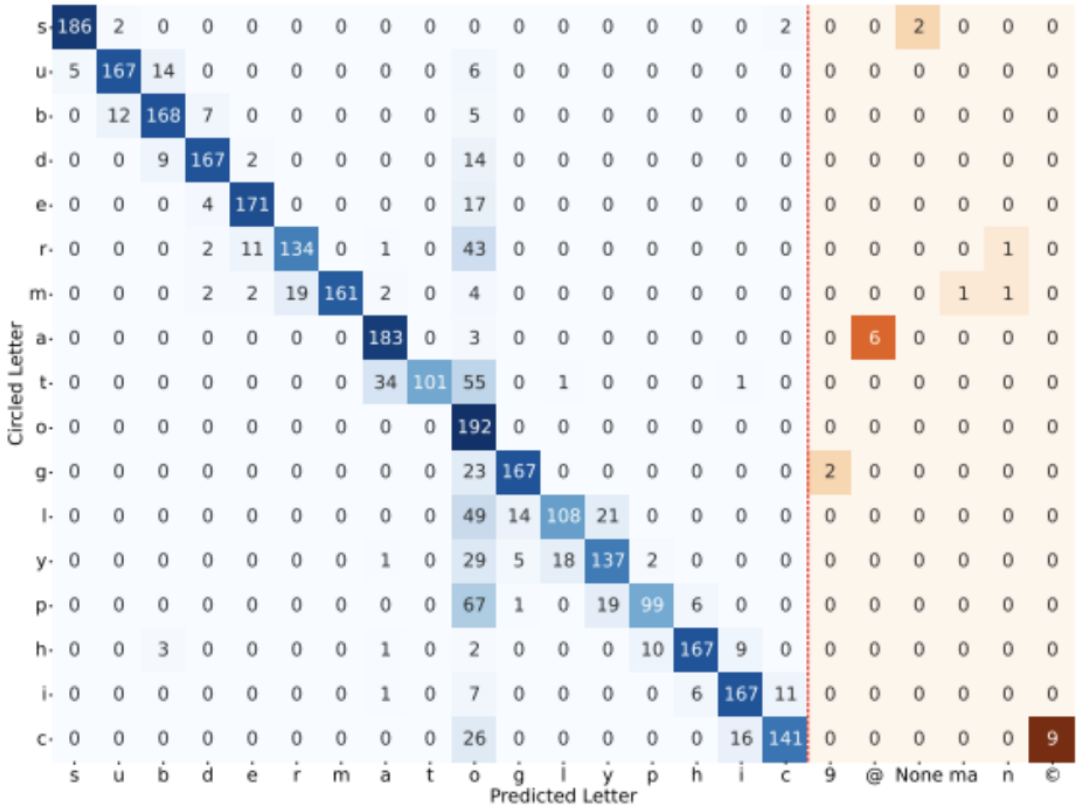


Fig. 10: Aggregate confusion matrix summed over all 4 VLMs and 48 images per model for the word Subdermatoglyphic. Each row sums to 192 (= 4 × 48). Models mostly mispredict to characters near the circled letter. VLMs also sometimes hallucinate characters that do not even exist in the word, *e.g.*, "9" or "n" (right panel).

Table 4: Except for GPT-4o, all other models have consistently higher accuracy (%) on the two English words than on the random string, suggesting that VLMs might leverage their familiarity with a known word to make educated guesses.

String					Mean
Acknowledgement	69.03	97.50	82.64	91.11	85.07
Subdermatoglyphic	63.60	91.05	71.45	94.49	80.15
tHyUiKaRbNqWeOpXcZvM	77.92	89.90	65.94	82.08	78.96
Mean accuracy	70.18	**92.81**	73.34	89.22	81.39

(표 4). VLMs는 두 프롬프트 (§C.5)와 두 글꼴 패밀리 (§C.4)에서 유사한 성능을 보입니다.

그림 10: 단어 "Subdermatoglyphic"에 대한 4개의 VLMs와 모델당 48개의 이미지를 모두 합한 총 혼동 행렬. 각 행의 합은 192입니다 (4 × 48). 모델은 주로 동그라미 친 문자 근처의 문자로 잘못 예측합니다. VLMs는 때때로 심지어 단어에 존재

하지 않는 문자(예: "9" 또는 "n")를 상상하기도 합니다 (오른쪽 패널).

표 4: GPT-4o를 제외한 다른 모든 모델은 무작위 문자열보다 두 개의 영어 단어에서 일관되게 더 높은 정확도(%)를 보이며, 이는 VLMs가 알려진 단어에 대한 친숙함을 활용하여 추론할 수 있음을 시사합니다.

String	GPT-4o	Gemini-1.5	Sonnet-3	Sonnet-3.5	Mean
Acknowledgement	69.03	97.50	82.64	91.11	85.07
Subdermatoglyphic	63.60	91.05	71.45	94.49	80.15
tHyUiKaRbNqWeOpXcZvM	77.92	89.90	65.94	82.08	78.96
Mean accuracy	70.18	92.81	73.34	89.22	81.39

이 데이터는 VLMs가 단어 인식에서 일정한 패턴을 보이며, 특히 알려진 단어에서 더 나은 성능을 보인다는 점을 보여줍니다.

4.4 VLMs는 겹치거나 중첩된 도형을 세는 데 어려움을 겪습니다

실험 우리는 모든 VLMs를 겹치는 원과 오각형 (§3.4) 및 중첩된 정사각형 (§3.5)의 모든 이미지에서 실행합니다. 모델에게 형식화된 답변으로 예측된 도형 수를 출력하도록 요청합니다. 추출된 답변을 정답과 비교합니다. 각 도형(원, 오각형, 정사각형)에 대해 두 가지 다른 프롬프트를 실행합니다.

결과 겹치는 원, 오각형 및 중첩된 정사각형을 세는 작업에서 VLM의 평균 정확도는 각각 51.35%, 30.99%, 73.29%입니다 (표 1d-f). 즉, 도형의 겹침이나 중첩 여부에 상관없이 도형을 세는 것은 모델에게 쉽지 않습니다. 즉, 가장자리들이 교차하거나 그렇지 않거나 (그림 11). 중첩된 정사각형의 경우, 모델의 정확도는 크게 다릅니다—GPT-4o (55.83%)와 Sonnet-3 (65.00%)는 Gemini-1.5 (80.00%)와 Sonnet-3.5 (92.08%)보다 적어도 30포인트 뒤처져 있습니다. 이 차이는 겹치는 원과 오각형을 세는 경우에 더 큽니다—Sonnet-3.5는 다른 모델보다 몇 배 더 좋습니다 (예: 77.29% 대 Sonnet-3의 9.79%; 표 5).

질문: 이미지에 있는 원의 개수는 몇 개입니까? 숫자 형식으로만 답하십시오.

그림 11: 겹치는 원을 세는 것은 VLMs에게 원의 색상, 선 너비 및 해상도에 관계없이 쉽지 않습니다. Gemini-1.5는 실제 원 수에 관계없이 종종 "5"를 예측하여 잘 알려진 올림픽 로고에 대한 강한 편향을 암시합니다.

모든 네 모델은 5개의 원을 세는 데 있어 최소 96%의 정확도를 보입니다. 그러나 원의 수를 하나씩 늘리면 Sonnet-3.5를 제외한 모든 모델에서 정확도가 거의 0에 가깝게 급격히 떨어집니다 (그림 12; 열 6-9). 오각형을 세는 경우, 모든 VLMs (Sonnet-3 제외)는 5개의 오각형에서도 성능이 저조합니다. 전체적으로, 6개에서 9개의 도형(원 및 오각형)을 세는 것은 모든 모델에게 어려운 과제입니다.

우리는 VLMs가 5개의 원을 세는 데 거의 완벽하게 수행하는 반면, 왜 5개 이상의 오각형 또는 도형을 세는 데 어려움을 겪는지를 추가로 조사합니다. 원이 5개 이상인 경우 (⚪) VLMs가 잘못된 수를 예측하면, Gemini-1.5는 실제 원의 수에 관계없이 "5"를 82.29%의 빈도로 예측합니다(표 6). 다른 모델의 경우 이 빈도는 오각형의 경우보다 훨씬 높습니다. 결과는 VLMs가 잘 알려진 5개의 원으로 이루어진 올림픽 로고에 편향되어 있음을 강하게 보여줍니다. 이러한 편향에 대한 자세한 내용은 §F.4에 있습니다.

표 5: 우리는 각 개수 하위 집합(예: 5, 6, 7, 8, 9개의 원)에 대한 별도의 정확도를 계산하고, 그런 다음 아래에 평균 정확도를 보고합니다(이러한 하위 집합 정확도를 기반으로). VLMs에게 중첩된 정사각형을 세는 것(a)이 겹치는 원(b)과 오각형(c)을 세는 것보다 쉽습니다. Sonnet-3.5는 세 가지 도형 모두에서 나머지보다 크게 우수한 성능을 보입니다.

Shape	GPT-4o	Gemini-1.5	Sonnet-3	Sonnet-3.5

Shape	GPT-4o	Gemini-1.5	Sonnet-3	Sonnet-3.5
(a) Circles	42.50	20.83	31.66	44.16
(b) Pentagons	17.50	19.37	9.79	77.29
(c) Squares	55.83	87.08	65.00	92.08

그림 12: 모든 네 VLMs는 5개의 원을 잘 셀 수 있습니다(왼쪽; 0.96), 하지만 Sonnet-3.5만이 0.92의 정확도로 5개의 겹치는 오각형을 잘 셀 수 있습니다(b). 6-9개 도형(원 또는 오각형)을 세는 것은 VLMs에게 도전적인 과제입니다. 흥미롭게도 GPT-4o와 Sonnet-3는 두 개의 중첩된 정사각형을 신뢰할 수 있게 세지 못합니다, 즉 0.88 및 0.91의 정확도입니다(c).

표 6: 원(⚪) 또는 오각형(⬠)이 5개 이상 있을 때 "5"를 예측하는 빈도(%), 즉 이미지에 6, 7, 8, 9개의 도형이 있는 경우. 예를 들어, Gemini-1.5는 실제 원의 수에 관계없이 82.29%의 빈도로 "5"를 예측하지만, 이 경향은 오각형의 경우에는 사라집니다(0%), 이는 5개의 원으로 이루어진 올림픽 로고에 대한 강한 편향을 보여줍니다(네 모델 중).

GPT-4o는 검은색 도형보다 색이 있는 도형에서 더 나은 성능을 보이며, Sonnet-3.5는 이미지 크기가 증가할수록 점점 더 나아집니다. 그러나 다른 모든 모델의 정확도는 색상 (§F.3) 및 이미지 해상도 (§F.2)를 변경할 때 다소 변화합니다.

각 이미지에는 중첩된 정사각형을 세는 작업에서 2개에서 5개의 정사각형만 있으며 이 정사각형들은 서로 교차하지 않습니다 (그림 13). 놀랍게도, GPT-4o와 Sonnet-3은 2개와 3개의 중첩된 정사각형을 완벽하게 세지 못합니다(그림 12c). 개수가 4개와 5개로 증가할 때, 모든 모델은 100% 정확도에서 멀어집니다(그림 12c). 우리의 결과는 모양의 경계가 교차하지 않을 때에도 VLMs가 정확한 모양의 표현을 추출하는 것이 쉽지 않음을 보여줍니다.

질문: 이미지에 몇 개의 정사각형이 있습니까? 중괄호 안에 숫자로 답하세요, 예: ({10}).

그림 13: 중첩된 정사각형을 세는 것은 VLMs에게 쉽지 않습니다. 정사각형이 두 개인 경우에도 그렇습니다(가장 왼쪽). 개수가 2개에서 5개로 증가할수록 작업은 더 어려워집니다. Sonnet-3.5가 가장 잘 수행하지만 (92.08%) 여전히 인간의 100%에는 미치지 못합니다.

4.5 VLMs는 그리드에서 행과 열을 쉽게 세지 못합니다

VLMs가 단순 도형의 개수를 세는 데 어려움을 겪기 때문에 (§3.4에서 도형의 경계가 교차할 때나 §3.5에서 분리될 때), 여기서는 이러한 도형들이 인접하여 가장자리를 공유하는 경우, 특히 여러 직사각형이 단일 그리드에 타일링될 때를 테스트합니다. DocVQA에서 표와 스프레드시트를 포함하는 질문에 대한 VLMs의 인상적인 정확도를 감안할 때 [4, 6, 36], 우리는 VLMs가 그리드에서 행과 열을 세어야 한다고 가정합니다.

실험 우리는 빈 그리드와 텍스트가 포함된 그리드 이미지를 모든 VLMs에 실행하고 그들의 형식화된 답변을 분석합니다 (§3.6).

결과 먼저, VLMs는 빈 그리드에서 행과 열을 세는 데 놀랍게도 형편없는 성과를 보였습니다(34.37% 정확도, 표 7 참조). 특히, 그들은 종종 하나 또는 두 개의 차이로 틀립니다(예: GPT-4o는 4x4를 예측하고 Gemini-1.5는 5x5를 예측, 4x5 그리드의 경우; 그림 14). 이 발견은 VLMs가 DocVQA에서 테이블 관련 질문에 답하기 위해 테이블에서 중요한 콘텐츠를 추출할 수 있지만 인간처럼 테이블을 셀별로 명확히 "볼" 수는 없음을 시사합니다.

이는 문서에서 표가 대부분 비어 있지 않기 때문에 VLMs가 사용되지 않기 때문일 수 있습니다. 이 가설과 일치하게, 단일 단어를 추가한 후 각 셀에 단어를 추가하면, 모든 VLMs의 정확도가 거의 두 배로 증가하는 것을 관찰할 수 있습니다(예: GPT-4o의 경우 26.13%에서 53.03%로) (표 7). 그러나, Sonnet-3.5가 텍스트가 포함된 그리드에서 88.68%로, 빈 그리드에서 59.84%로 가장 잘 수행하는 모델로 이 작업을 해결할 수 있는 모델은 없습니다(그림 15a vs b).

질문: 행과 열의 수를 세고 중괄호 안에 숫자로 답하세요. 예를 들어, rows=({5}) columns=({6}).

그림 14: VLMs는 빈 그리드에서 행과 열을 세는 데 종종 한두 개의 오차를 냅니다. 그리드가 작고(예: 3x4) 각 셀에 단어가 포함될 때도 마찬가지입니다.

표 7: 그리드 내에 텍스트를 포함하면 모든 모델의 정확도가 향상됩니다. Sonnet-3.5는 빈 그리드와 텍스트가 포함된 그리드 모두에서 다른 모델보다 뛰어난 성능을 보입니다.

Grid	GPT-4o	Gemini-1.5	Sonnet-3	Sonnet-3.5	Mean
Empty	26.13	26.51	25.00	59.84	34.37
Text	53.03	52.27	47.34	88.68	60.33
Mean	39.58	39.39	36.17	74.26	47.35

표 8: VLM의 정확도(%)는 경로의 수가 1에서 3으로 증가함에 따라 감소합니다. 전체적으로, 색깔이 있는 경로를 세는 것은 VLMs에게 도전 과제를 제시합니다.

Paths	GPT-4o	Gemini-1.5	Sonnet-3	Sonnet-3.5	Mean
1	56.25	64.58	22.91	92.91	59.16
2	47.44	38.35	28.69	48.29	40.69
3	40.00	21.87	18.12	25.41	26.35
Mean	47.89	41.60	23.24	55.53	42.06

흥미롭게도, VLMs는 행보다 열을 세는 데 더 능숙합니다 — 70.53% 대 60.83% 정확도 (그림 15c vs d). 그러나, 이 숫자들은 여전히 100%에서 멀리 떨어져 있어, VLMs가 현재 테이블에서 행이나 열을 신뢰할 수 있게 세지 못한다는 것을 보여줍니다. 더 많은 결과는 §§ G.2 및 G.3을 참조하십시오.

4.6 VLMs는 단일 색상의 경로를 세는 데 어려움을 겪습니다

이 경로 세기 작업은 VLM의 고유 색상의 경로를 인식하고 시작 지점에서 목적지까지 추적하는 능력을 테스트합니다. 이는 일반적으로 지도나 그래프를 읽는 데 중요한 작업입니다 [29].

그림 15: 그리드가 비어 있을 때(a)와 텍스트를 포함할 때(b) 행(c) 대 열(d)을 세는 정확도(여기서, 개별적으로 분석됨). VLMs(특히, Gemini-1.5와 Sonnet-3)는 일반적으로 그리드에 텍스트가 포함된 경우(b)가 빈 그리드(a)보다 훨씬 더 정확하게 셉니다. 흥미롭게도, 열(d)은 행(c)보다 VLMs가 세기 더 쉽습니다.

실험 지하철 지도에서 (§3.7) 우리는 두 개의 연결된 역을 무작위로 선택하고 모든 모델에게 해당 역을 연결하는 단일 색상의 경로를 세도록 요청합니다. 우리는 VLM 템플릿 응답에서 숫자를 추출하고 이를 정답과 비교합니다.

결과 전반적으로, VLMs는 평균 정확도 42.06%로 성능이 저조합니다(표 1h). 두 역 사이에 경로가 하나밖에 없는 경우에도, Sonnet-3.5가 92.91%로 최고이지만, 최악은 22.91%로(표 8) 100% 정확도에 도달할 수 있는 모델은 없습니다. VLM이 예측한 카운트는 종종 1에서 3 경로 차이가 납니다(그림 16). VLM 정확도는 경로의 복잡성이 1에서 2, 3으로 증가함에 따라 각각 59.16%에서 40.69% 및 26.35%로 크게 감소합니다(표 8). VLM 응답의 더 많은 샘플은 §H.2에 있습니다.

---

5 관련 연구

VLM 비전 이해 벤치마킹: 대학 수준의 주제 [47], 차트 [29], 문서 [30] 또는 비디오 [46]는 VLM 비전 이해를 평가하기 위한 일반적인 벤치마크 중 일부입니다 [4, 6, 10, 36]. VLM의 최근 급속한 발전을 목격하고 있습니다. 예를 들어, Sonnet-3.5

는 DocVQA에서 95.2%, ChartQA에서 90.8%, AI2D에서 94.7%에 도달하고 있습니다 [6]. 그러나 대부분의 비전 벤치마크는 실제 세계, 주제별 데이터에서 VLMs를 평가하려고 시도하며, 이는 광범위한 사전 지식이 필요하며 "데이터 누출" 문제를 가지고 있습니다. 즉, VLMs는 입력 이미지 없이도 정확하게 답할 수 있습니다 [12]. 더욱이, 대부분의 벤치마크는 인간-기계 지능 격차에 대한 고차원의 감각을 제공하기 위해 인간이 처리해야 하는 데이터에 대해 VLMs를 테스트합니다 [25, 45]. 그러나 우리의 BlindTest 벤치마크는 이전 벤치마크와 크게 다릅니다. 이는 (1) 인간에게 매우 쉽고 5세 어린이가 해결할 수 있으며(차트 [29, 30, 47]와 달리); (2) VLMs를 위한 첫 번째 저수준의 시각적 검증 검사; (3) 최소한의 사전 지식을 요구함; (4) 최소한의 상식을 요구함.

질문: A에서 C로 가는 단일 색상의 경로가 몇 개 있습니까? 중괄호 안에 숫자로 답하세요. 예: ({3}).

그림 16: 일부 VLMs (GPT-4o, Gemini-1.5, Sonnet-3)는 놀랍게도 두 선 너비(가장 왼쪽) 모두에서 극도로 쉬운 경우에 실패합니다. VLMs는 역을 연결하는 경로의 수가 증가할수록 성능이 저하되는 경향이 있습니다.

복잡한 추론과 달리(예: [13, 48]), 강력한 언어 모델은 BlindTest 이미지에서 언어로 설명하는 것이 사람에게 자연스럽지 않을 때는 거의 쓸모가 없습니다.

ARC 벤치마크 [13, 32]는 추상 이미지를 포함하고 있으며, 간단한 모양으로 구성되어 있습니다. 그러나 이는 VLMs에게 그 패턴을 기반으로 이해하고 추론하도록 도전합니다. 즉, ARC는 VLMs가 추론하기 위해 추상적인 모양을 식별할 수 있다고 가정합니다. 반면에, 우리의 BlindTest는 이러한 원시 모양을 인식하는 VLM의 능력을 직접 평가합니다.

VLM 시각적 능력 향상

최근의 최첨단 VLMs를 향상시키는 대부분의 방법은 고수준의 시각적 작업을 해결하기 위해 사전 훈련된 LLM에 비전 인코더를 결합하여 미세 조정하는 것입니다 [23]. 이러한 후반 융합 접근법은 토큰화된 이미지에서 학습한 시각적 표현을 강력한 사고 체계와 융합합니다 [21, 22, 31].

그러나 현재의 VLMs를 위한 시각적 접근법은 모델이 실제 사진에 존재하는 자연 물체를 볼 수 없는 경우 "맹목적"인 것처럼 도전 과제를 직면하고 있습니다 [41]. 반면에, 우리는 VLMs가 저수준의 추상 이미지에서 시각적으로 손상된 상태임을 보여줍니다. 예를 들어, 6개의 겹치는 원 또는 3개의 중첩된 사각형을 세는 능력의 부재 (§3.3)는 자연 이미지에서 실제 물체와의 상호작용을 인식하는 VLM의 능력에서 영감을 받았습니다 [37, 43, 44]. 그러나 우리는 VLMs가 실제 물체를 인식하는 것과는 반대로 저수준의 광학 문자 인식에서 실패할 수 있음을 보여줍니다. 우리가 아는 한, 우리의 논문에서 제기된 정확한 제한 사항을 해결하려는 시도는 없었습니다: (1) 경로, 모양 및 기하학적 도형을 식별하고 계산하는 것, 특히 이들이 상호작용할 때 (§4.1 to §4.5); (2) 색상 경로를 따르는 것 (§4.6). 이러한 제한 사항을 해결하는 것은 기존의 시각적 벤치마크에서 VLMs가 발전할 수 있는 기반이 될 수 있습니다.

질문: 두 원 사이의 겹치는 영역의 색은 무엇입니까?

- **GPT-4o:** 두 원 사이의 겹치는 영역은 터키색(시안의 한 색조)입니다.
- **Gemini-1.5:** 녹색 원과 파란색 원의 겹치는 영역은 **시안**일 것입니다.
- **Sonnet-3:** 이 이미지의 두 원 사이에는 겹치는 영역이 없습니다. 이미지는 서로 교차하거나 겹치지 않는 두 개의 별도 원 - 녹색 원과 파란색 원 - 을 보여줍니다. ❌
- **Sonnet-3.5:** 죄송합니다, 이 이미지의 두 원 사이에는 겹치는 영역이 없습니다. 이미지는 두 개의 뚜렷한 원 - 하나는 녹색이고 다른 하나는 파란색 - 을 보여줍니다. ❌

그림 17: 때때로 VLMs(GPT-4o와 Gemini-1.5)는 명확히 보지 못하고 추측하여 답을 합니다.

---

6 논의 및 결론

우리는 BlindTest라는 벤치마크를 제안합니다. 이는 선, 원, 사각형, 교차점과 같은 기본적인 기하학적 원시 요소를 "볼" 수 있는 VLM의 능력을 테스트하는 일곱 가지 새로운 저수준 시각적 작업의 벤치마크입니다. 이 작업들은 처음부터 설계되었으며 최소한의 사전 지식을 요구합니다. 이러한 작업은 인터넷에 존재하지 않았고 최소한의 세계 지식을 필요로 하므로 VLMs가 암기로 BlindTest를 해결할 가능성은 거의 없습니다. 입력 이미지를 사용하지 않는 문제는 이전의 일부 벤치마크에서 문제였습니다 [12, 16].

우리는 최고의 상업용 VLMs도 다섯 살짜리가 쉽게 해결할 수 있는 작업(예: 두 원 또는 선이 교차하는지 여부 판단)에서 여전히 어려움을 겪고 있음을 놀랍게도 발견했습니다 (§4.2). 이러한 제한은 LLM 모델에 시각을 통합하는 후반 융합 접근법 [9, 24] 때문일 수 있으며, 초기 융합 [39, 40]이 향후 해결책이 될 수 있습니다. 우리는 LoRA를 사용하여 7B, 최첨단 개방형 VLM [15]을 미세 조정했지만 성능이 좋은 모델을 얻지 못했습니다 (§B.6). 즉, BlindTest에서 잘 수행하는 일반 VLM을 훈련하는 것은 비트리비얼하고 흥미로운 연구 방향일 수 있습니다.

BlindTest에서 VLMs의 저조한 성능은 모델이 화살표 방향이나 경로를 따라야 하는 실제 시각적 작업에서 저조할 수 있음을 시사합니다(예: 그림 45의 지하철 지도 읽기, 특정 경로 또는 방향을 필요로 하는 지도, 그림 46의 그래프 읽기). 프롬프트를 보지 않고 시각적 특징을 추출하기 때문에, VLMs는 일부 저수준 세부 사항을 "보지 못할" 수 있으며, 이로 인해 지능형 언어 모델이 어려움을 겪고 때때로 추측을 하게 될 수도 있다.