Vision language models fail to translate detailed visual features into words

Pooyan Rahmanzadehgervi† pooyan.rmz@gmail.com
Logan Bolton† logan.bolton@auburn.edu
Mohammad Reza Taesiri♢ mtaesiri@gmail.com
Anh Totti Nguyen† anh.ng8@gmail.com
† Auburn University ♢ University of Alberta

Abstract

Recent studies demonstrate that large language models with vision capabilities (VLMs), e.g., *GPT-4o and Gemini-1.5 Pro, struggle with low-level vision tasks that are easy to humans. Specifically, on BlindTest, the suite of 7 very simple tasks, including identifying (a) whether two circles overlap; (b) how many times two lines intersect; (c) which letter is being circled in a word; and (d) the number of circles in an Olympic-like logo, four state-of-the-art VLMs are only 58.07% accurate, on average. In this work, we investigate the potential reasons behind this phenomenon. We find that VLMs, including slow-thinking models, consistently struggle with those tasks that require precise spatial information when geometric primitives overlap or are close. Yet, VLMs perform at near-100% accuracy when much more space is added to separate shapes and letters. Linear probing experiments show that vision encoders contain sufficient visual information to solve BlindTest and that language models fail to decode this information into correct answers.*

1. Introduction

Exploring the visual shortcomings of large language models with vision capabilities (VLMs) has revealed surprising findings recently [15, 18, 19]. Specifically, VLMs struggle to perform at near-human accuracy on simple visual tasks, *e.g.*, the best performing VLM on BlindTest [15], *i.e.*, Sonnet-3.5, performs at 77.84% accuracy. Since common image-text benchmarks [8, 11, 21] fail to capture VLMs' true visual capabilities [4], we hypothesize that these shortcomings lie in their visual perception abilities.

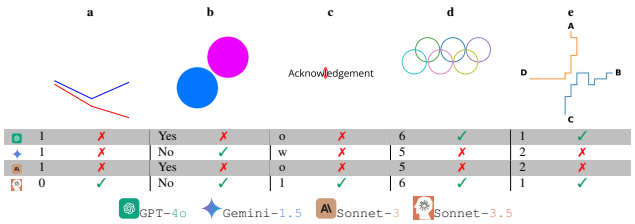
In this work, we aim to explore the low-level visual abilities of VLMs inspired by [15]. Specifically, we test four state-of-the-art (SotA) VLMs: GPT-4o [12], Gemini-1.5 Pro [16], Claude-3 Sonnet [3], and Claude-3.5 Sonnet [2] on simplified versions of tasks from BlindTest [15], which includes simple visual tasks that involve only 2D geometric primitives (*e.g.*, lines and circles) [6] and require minimal world knowledge. We also investigate the impact of slow-thinking capabilities on vision-centric tasks from [15]. Our findings show that the reasoning process in these models is not sufficient to perform better on primitive visual tasks. Moreover, we conduct a linear probe experiment on 2 tasks from BlindTest, *i.e.*, a) counting line intersection and b) identifying whether two circles touch, using open-source VLMs to find if the necessary information for solving these tasks exists at various stages of the VLM. Similar to the human visual cognitive system [9, 13], VLMs follow a common late-fusion architecture, connecting "eyes", *i.e.*, pre-trained vision encoders, to a talking brain, *i.e.*, a pre-trained LLM (see Fig. 5). Thus, we contrast our findings at each stage in VLMs with the reported performance on BlindTest to test our hypothesis.

2. Experiments and Results

We evaluate VLMs in 3 different categories: (1) commercial, (2) open-source, and (3) slow-thinking. In commercial models, we test GPT-4o (), Gemini-1.5 Pro (Gemini-1.5), Claude-3 Sonnet (Sonnet-3), and Claude-3.5 Sonnet (Sonnet-3.5).

Open-source We also test 8 *open-source* models from three different families: LLaVA OneVision-qwen2 (LLaVA-OneV) [10], Phi-3.5-vision-instruct (Phi-3.5) [1], and InternVL-2 ( ) [5].

Slow-thinking We test 2 slow-thinking models and their regular counterparts: (1) closed-source Gemini 2.0 Flash-Thinking and Gemini 2.0 Flash, (2) open-source QVQ-Preview (QVQ) and Qwen2-VL [17, 20], to evaluate the impacts of slow and iterative thinking on BlindTest.



a: How many times do the blue and red lines touch each other? Answer with a number in curly brackets, e.g., {5}.
b: Are the two circles overlapping? Answer with Yes/No.
c: Which character is being highlighted with a red oval? Please provide your answer in curly brackets, e.g., {a}
d: How many circles are in the image? Answer with only the number in numerical format.
e: How many single-color paths go from A to D? Answer with a number in curly brackets, e.g., {3}.

Figure 1. VLMs fail on the simple tasks of BlindTest.

Table 1. The mean accuracy (%) of all closed-source VLMs over 7 BlindTest tasks is 58.07%. * Two smallest VLMs used in linear probing experiments.

Model	Size	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
GPT-4o	n/a	41.61	75.91	74.23	41.25	20.21	55.83	39.58	53.19		50.23
Gemini-1.5	n/a	66.94	**93.62**	83.29	20.25	24.17	87.08	39.39	53.13		58.48
Sonnet-3	n/a	43.41	86.46	72.06	29.79	1.87	65.00	36.17	31.11		45.73
Sonnet-3.5	n/a	**75.36**	90.82	**87.88**	**66.46**	**77.71**	**92.08**	**74.26**	**58.19**		**77.84**
Mean		56.84	86.70	79.36	39.44	30.99	74.99	47.35	48.90		58.07
Open-souce VLMs											
LLaVA-OneV-ov	72B	45.83	90.92	44.71	20.00	11.74	87.07	8.95	58.06		45.92
LLaVA-OneV-si	72B	45.33	83.48	38.14	20.00	11.46	57.50	10.23	48.06		38.41
LLaVA-OneV-ov	7B	48.17	83.93	42.79	20.00	7.29	42.92	21.02	47.22		39.17
LLaVA-OneV-si	7B	44.50	84.67	40.22	20.00	7.29	58.75	14.01	55.00		40.00
LLaVA-OneV-ov	0.5B	17.28	75.07	9.78	12.50	9.58	20.42	0.38	5.56		18.82
LLaVA-OneV-si *	0.5B	33.14	73.21	6.25	27.29	2.50	14.58	1.13	26.11		23.03
InternVL-2	8B	47.28	91.00	57.69	20.00	13.96	28.33	7.57	60.28		40.76
Phi-3.5 *	4.2B	37.78	83.63	16.51	18.75	11.46	32.50	11.74	19.72		29.01

2.1. VLMs read out circled letter better when spacing between letters increases

Rahmanzadehgervi et al. [15] find that reading out which letter is being circled (Fig. 1c) in a word is a challenging task for VLMs (mean model accuracy: 79.7%; Tab. 1c). They report that **When the letters are close together, VLMs often predict letters adjacent to the one being circled** (Fig. 1c).

Here, we add 1 to 3 ASCII space characters between adjacent letters of a string (Fig. 2a) to evaluate VLMs to test the adjacency hypothesis. All VLMs consistently perform better when there is $\geq$1 extra space between characters (Fig. 2b). However, the accuracy increase ($\triangle$) varies across models. For instance, and accuracies increase by over +20 points to 92% and 72% from 72% and 46%, respectively (Fig. 2b). Out of 4 tested VLMs, Sonnet-3.5 reaches the highest accuracy of 95% when there are 3 extra spaces between letters (Fig. 2a). Qualitatively, the remaining 5% error (40 samples) includes: (1) 12 mispredictions of adjacent letters and (2) 13 instances of confusing the red circle as part of the letter, *e.g.*, '@' for 'a', and (3) 15 cases of predicting 'g' instead of 'q'.

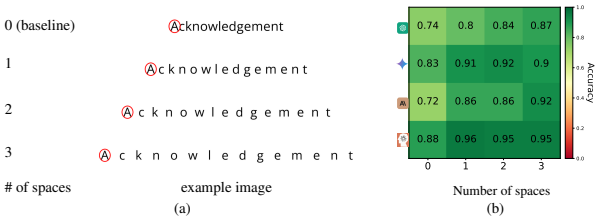


Figure 2. (a) By adding more space $\in \{1, 2, 3\}$ to baseline images in the circled letter task, we create a simpler version of them. (b) VLMs generally perform better when there is $\geq$1 space between letters of the words ($\triangle$ for Sonnet-3 and GPT-4o is +21 and +13 from 0 to 3 spaces, respectively).

2.2. VLMs count shapes more accurately when more spacing is added between shapes

On counting overlapping circles and pentagons, in BlindTest, VLMs' mean accuracy is 39.44% and 30.99% respectively (Tab. 1d–e). Inspired by the results that VLMs read out circled letters better when there is more space between letters, here, we study the effects of increasing the space between shapes, in counting overlapping shapes, for both and (Fig. 1d). We test whether reducing the overlap area between shapes would improve VLM accuracy in counting them. Specifically, we increase the boundary-to-boundary distance between circles in the original images along the X and Y directions is $\in \{dx \times \frac{\phi}{2}, dy \times \frac{\phi}{2}\}$, where ϕ is the diameter of the circles and dx and dy are multipliers (see Fig. 3a). For circles, we increase dx and dy. For pentagons, we increase the boundary-to-boundary distance of $\{d \times dx, d \times dy\}$ where d is the side length of the pentagons (see Fig. 3b).

VLMs, in general, can count shapes more accurately when there is no overlapping area between shapes (Fig. 3c). Yet, the accuracy increases vary between models. For example, both Sonnet-3 and Sonnet-3.5 reach $\geq$96% (Fig. 3c; $dx = 0.75$). Similarly, 72B-LLaVA-OneV () achieves 72% accuracy on counting disjoint circles (Fig. 3c; $dx = 0.75$). This shows that most VLMs struggle to count the shapes in the baseline images (Fig. 3c; $dy = -1$ and $dx = 0.1$) because they overlap. All closed-source VLMs, except for Gemini-1.5, consistently benefit from increasing the distance along both directions in counting overlapping shapes. The most significant improvement is for Sonnet-3 with $\triangle$=91% and the least is for GPT-4o with $\triangle$=22% (Fig. 3c-ii).

2.3. VLMs can count simplified, more straight paths

Overall, VLMs perform poorly at a mean accuracy of 48.90% (Tab. 1h) on counting the single-colored paths. Here, we investigate whether VLMs are not able to count in general or whether the zigzag patterns of paths (Fig. 1e) pose the main challenge to them.

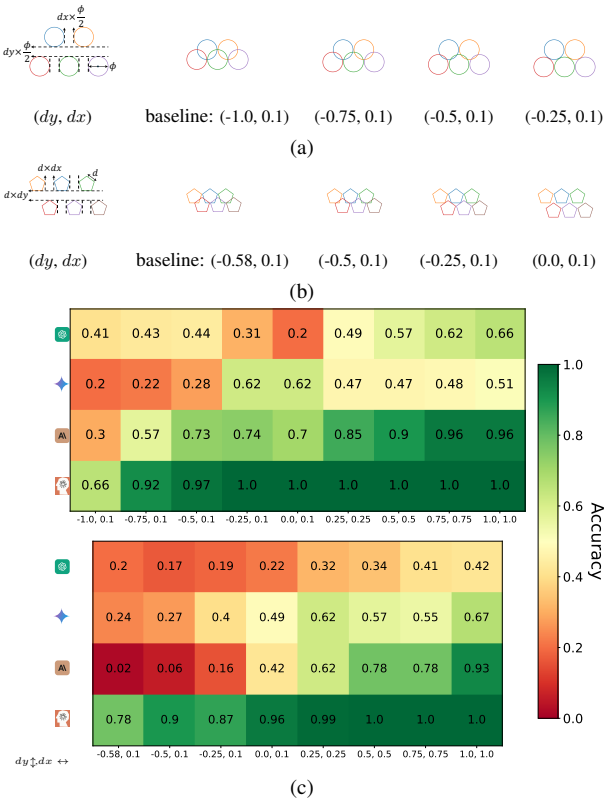


Figure 3. (a) We reduce the overlap area between circles by increasing the boundary-to-boundary distances along the X and Y axes, *i.e.*, $dx \times \frac{\phi}{2}$ and $dy \times \frac{\phi}{2}$, respectively. (b) We increase the boundary-to-boundary distances between pentagons along the X and Y axes, *i.e.*, $d \times dx$ and $d \times dy$, respectively. (c) As we increase the distance between the shapes along the X and Y axes for both circles and pentagons, VLMs' accuracy improves. For example, `Sonnet-3.5` ( ) accuracy increases (i) from 0.66 to 1.0 on and (ii) from 0.78 to 1.0 on .

We re-render the images by forcing each path to have fewer 90° turns than the baseline. The baseline images (Fig. 1e) are generated by choosing a direction on a grid using a random depth-first algorithm, where the probability (P) of choosing a straight direction is 0.33. Therefore, we gradually increase the P from the baseline (P=0.33) to 0.6 and 0.9, such that it yields images with fewer intersections and turns (Fig. 4a).

On average, all VLMs more accurately count the single-colored paths when there are fewer turns, *i.e.*, as P increases (see Fig. 4b). This indicates that SotA VLMs mostly struggle to count the paths in original images due to the visual complexity of zigzag patterns of paths and their intersection. Analyzing the accuracy by the number of paths connected to each station, we find that some VLMs even score near-100 accuracy (*e.g.*, 0.95, 0.99, and 0.95 for , , and , respectively, at $P = 0.9$; Fig. 4). This substantially

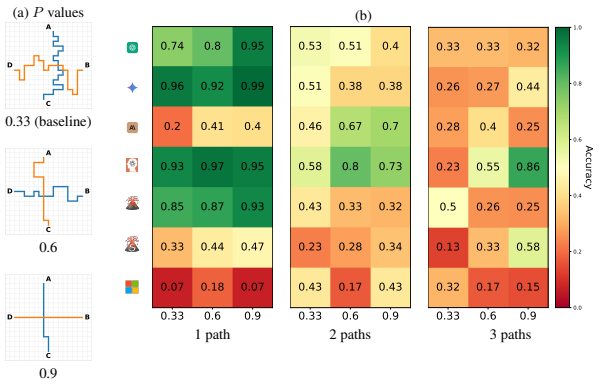


Figure 4. We increase the probability P of choosing a straight next move (as opposed to making a turn) and generate two simplified versions of subway-like maps (a). As we increase the probability P from 0.33 to 0.6 and 0.9 (b), some VLMs can reach a near-perfect accuracy (0.99 for `Gemini-1.5` on 1 path).

better accuracy on simplified images is in stark contrast to the poor accuracy reported for the original subway maps ($P = 0.33$), confirming that the visual complexity of the paths poses challenges to VLMs.

2.4. The vision encoders in smaller open-source VLMs can extract sufficient information to solve `BlindTest`

Here, we question whether VLMs can "see" the key visual information in `BlindTest` images, *e.g.*, the gap between two circles in order to decide whether they overlap. Specifically, we run linear probing experiments to test whether the visual encoders of the smallest open-source VLMs can extract sufficient information for solving `BlindTest`.

Models We select two models from SotA open-source VLM, 0.5B `LLaVA-OneV-S` ( ) and 4.2B `Phi-3.5` ( ) for **two reasons**. First, these two VLMs use the two most common vision encoders (VEs)— uses SigLIP [22] while uses CLIP [14]. That is, our findings on these two VEs would generalize to most VLMs. Second, and are among the smallest VLMs, and therefore, if their VEs contain sufficient information, the same is likely true with larger and commercial VLMs.

Tasks We choose (1) the two circles ( ) and (2) the counting line-intersections ( ) for this experiment because they represent arguably the simplest images and questions in `BlindTest`—the two circles and the line intersections tasks are 2-way and 3-way classification problems.

Method We average-pool the image-patch features at the layer right *before* the projection layer (Fig. 5). Then, we

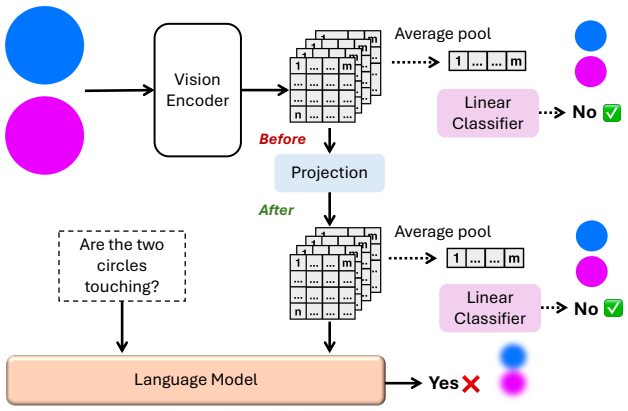


Figure 5. We train a linear-probing classifier on the frozen features extracted from the (1) vision encoder and the (2) projection layer for the two circles and the line chart tasks separately. Evaluating the linear classifiers shows that the information necessary to solve these two tasks exists before and after the projection layer, but is lost in the LLM, resulting in poor VLM accuracy in Tab. 1.

train a logistic-regression linear classifier on top of the frozen features on each task. For completeness, we repeat the experiment for the layer right *after* the projection layer to understand the impact of the projection layer.

Results The linear-probing accuracy of the CLIP features, **before projection** layer, in is $\geq$ 99.47% on both tasks (Tab. 2). This suggests that the necessary low-level information to solve these tasks is preserved in CLIP. Similarly, the same conclusion holds for the VE in , a variant of SigLIP, that performs 100% on both tasks (Tab. 2). Moreover, using the frozen features **after the projection** layer in both and yields a linear classification accuracy of $\geq$ 99.58% on both tasks (Tab. 2). This result shows that most visual information from VEs is preserved before and after the projection layer. Contrasting these high linear-classification VE accuracy scores with fairly lower accuracy of both VLMs on `BlindTest` (see Tab. 2; rightmost column), we conclude that the **language models in these VLMs have access to the necessary visual information to solve `BlindTest` tasks but fail to decode it into correct language outputs**.

2.5. Long-inference, slow-thinking VLMs also perform poorly like regular VLMs on `BlindTest`

From math to coding, spending more time thinking before responding enables LLMs to perform substantially better in many tasks [7]. Here, we aim to test whether such slow thinking also enables VLMs to perform better on `BlindTest` where we argue that reasoning in the text

Table 2. The output features from the vision encoders right before (a) the projection layer in `LLaVA-OneV-S` () and `Phi-3.5` (), *i.e.*, CLIP and SigLIP, respectively, contain sufficient information to solve the and (linear-probing accuracy is $\geq$ 99.47%). The same conclusion holds for after (b) the projection layer. However, the language model in these VLMs fails to decode this information into correct answers, resulting in poor accuracy on the tasks (c).

	(a) Before		(b) After		(c) VLM	
Model						
	99.47	99.82	99.58	99.73	33.14	73.21
	100.0	100.0	100.0	100.0	37.78	83.63

Table 3. SOTA slow-thinking models (bottom) perform even worse than their regular counterpart (top) `BlindTest`, showing that the longer inference has no positive impact on `BlindTest` tasks. `QVQ` is the slow-thinking counterpart of `Qwen2-VL`.

Model	Size									Task mean
`Gemini 2.0 Flash`	n/a	**85.44**	80.95	**81.73**	**55.62**	44.16	**96.25**	66.85	**70.97**	**72.75**
`Qwen2-VL`	72B	64.97	76.41	73.56	28.12	35.42	74.58	20.64	59.31	54.13
`Gemini 2.0 Flash-Thinking`	n/a	77.50	**88.24**	74.03	47.92	**57.08**	93.75	**70.45**	63.75	71.59
`QVQ`	72B	37.05	67.26	51.60	29.58	26.66	53.75	36.74	37.22	42.48

space might not help as our tasks are, by design, visual only. We run 2 SOTA slow-thinking VLMs: `Gemini 2.0 Flash-Thinking` and `QVQ` on `BlindTest`, and compare them with their non-thinking, regular versions, *i.e.*, `Gemini 2.0 Flash` and `Qwen2-VL`.

On average, over 7 tasks, `Gemini 2.0 Flash-Thinking`, is on par with its non-thinking counterpart, `Gemini 2.0 Flash` (Tab. 3; 71.59 vs 72.75%). This shows that the "slow-thinking" capability (*i.e.*, long, scaled-up inference) does not address the main challenge that `BlindTest` poses to VLMs. Qualitatively examining the thinking tokens of `Gemini 2.0 Flash-Thinking` shows that the hidden thoughts are in text space and have no benefits on `BlindTest`. Similarly, `QVQ`, the SOTA open-source slow-thinking model, is -11.65 points behind its non-thinking counterpart, `Qwen2-VL` (Tab. 3).

3. Discussion and Conclusion

We explore the reasons behind VLMs' poor performance on simple visual tasks from `BlindTest` [15]. We generate simpler versions of the tasks by gradually reducing the visual complexity of the images, *e.g.*, overlapping area, space between letters, and number of turns and intersections. Our findings show that as the images get more crowded, VLMs fail more often. Moreover, by conducting a linear probe experiment on 2 common vision encoders of VLMs, we find that the information necessary to solve these tasks is preserved in their representation, and the language model fails to translate it to words.

References

[1] Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Harkirat Behl, et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024. 1

[2] Anthropic. Introducing claude 3.5 sonnet \ anthropic. https://www.anthropic.com/news/claude-3-5-sonnet. (Accessed on 07/03/2024). 1

[3] AI Anthropic. The claude 3 model family: Opus, sonnet, haiku. *Claude-3 Model Card*, 2024. 1

[4] Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, et al. Are we on the right way for evaluating large vision-language models? *arXiv preprint arXiv:2403.20330*, 2024. 1

[5] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. *arXiv preprint arXiv:2312.14238*, 2023. 1

[6] John F Hughes. *Computer graphics: principles and practice*. Pearson Education, 2014. 1

[7] Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024. 4

[8] Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 235–251. Springer, 2016. 1

[9] Doeon Lee, Minseong Park, Yongmin Baek, Byungjoon Bae, Junseok Heo, and Kyusang Lee. In-sensor image memorization and encoding via optical neurons for bio-stimulus domain reduction toward visual cognitive processing. *Nature Communications*, 13(1):5223, 2022. 1

[10] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024. 1

[11] Ahmed Masry, Xuan Long Do, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2263–2279, Dublin, Ireland, 2022. Association for Computational Linguistics. 1

[12] OpenAI. Hello gpt-4o | openai. https://openai.com/index/hello-gpt-4o/. (Accessed on 05/31/2024). 1

[13] Zhuiri Peng, Lei Tong, Wenhao Shi, Langlang Xu, Xinyu Huang, Zheng Li, Xiangxiang Yu, Xiaohan Meng, Xiao He, Shengjie Lv, et al. Multifunctional human visual pathway-replicated hardware based on 2d materials. *Nature Communications*, 15(1):8650, 2024. 1

[14] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. 3

[15] Pooyan Rahmanzadehgervi, Logan Bolton, Mohammad Reza Taesiri, and Anh Totti Nguyen. Vision language models are blind. In *Proceedings of the Asian Conference on Computer Vision*, pages 18–34, 2024. 1, 2, 4

[16] Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy Lillicrap, Jean-baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024. 1

[17] Qwen Team. Qvq: To see the world with wisdom, 2024. 1

[18] Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, et al. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. *arXiv preprint arXiv:2406.16860*, 2024. 1

[19] Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9568–9578, 2024. 1

[20] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 1

[21] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of CVPR*, 2024. 1

[22] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11975–11986, 2023. 3