This CVPR Workshop paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Interpreting COVID Lateral Flow Tests' Results with Foundation Models

Stuti Pandey[1], Josh Myers-Dean[1], Jarek Reynolds[1], Danna Gurari[1,2]
[1]University of Colorado Boulder, [2]The University of Texas at Austin

Abstract

Lateral flow tests (LFTs) enable rapid, low-cost testing for health conditions including Covid, pregnancy, HIV, and malaria. Automated readers of LFT results can yield many benefits including empowering blind people to independently learn about their health and accelerating data entry for large-scale monitoring (e.g., for pandemics such as Covid) by using only a single photograph per LFT test. Accordingly, we explore the abilities of modern foundation vision language models (VLMs) in interpreting such tests. To enable this analysis, we first create a new labeled dataset with hierarchical segmentations of each LFT test and its nested test result window. We call this dataset LFT-Grounding. Next, we benchmark eight modern VLMs in zero-shot settings for analyzing these images. We demonstrate that current VLMs frequently fail to correctly identify the type of LFT test, interpret the test results, locate the nested result window of the LFT tests, and recognize LFT tests when they partially obfuscated. To facilitate community-wide progress towards automated LFT reading, we publicly release our dataset at `https://iamstuti.github.io/lft_grounding_foundation_models/`

1. Introduction

Vision-language models (VLMs) have demonstrated impressive zero-shot capabilities in describing images. This development begs a question as to how far such models' abilities extend. We explore VLMs' abilities for a critical medical problem: analyzing Lateral Flow Tests (LFTs) [5]. An LFT [5] is a cost-effective diagnostic tool for rapidly identifying health conditions.

Our work contributes to the growing interest in automating LFT analysis [11, 21, 31]. The motivation for automated interpretation of LFT results are numerous. For instance, such a solution could empower blind people to independently learn about their health [9, 10, 19, 25], thereby broadening accessibility and inclusivity in healthcare diagnostics. Automated readers could also accelerate data entry for large-scale monitoring (e.g., for pandemics such as COVID-19) by only requiring the capture of a single photograph per LFT test [5].

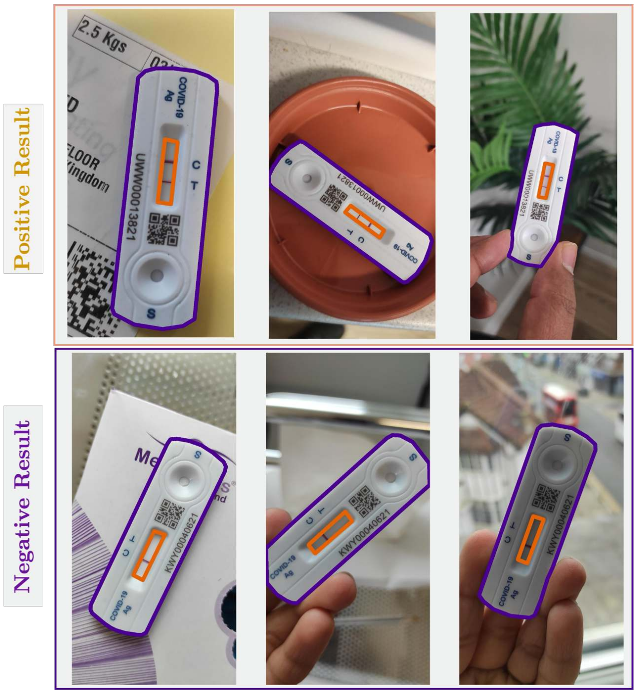


Figure 1. Examples from our dataset of images showing COVID-19 LFTs with positive results *(first row)* and negative results *(second row)*. We introduce segmentations of each LFT test (indicated in purple) and its test result window (indicated in orange).

Our first contribution is an LFT-based dataset to enable evaluating models' predictive performance. Extending prior work [4, 17, 29], we introduce the first dataset that locates the visual evidence used to arrive at a test result interpretation. Specifically, for a collection of images showing Covid LFTs [17], we augment each image's label indicating whether the result is positive or negative with segmentations of the test and its nested test result window. We call the resulting dataset **LFT-Grounding** to it provides **groundings** that localize LFT tests and their nested test result windows. Examples of annotated images are shown in Figure 1.

Our next contribution is to benchmark eight modern VLMs that generate image captions in zero-shot settings to see how effectively they recognize the image contents and reason to those descriptions based on the correct visual evidence. We find that existing VLMs often struggle to recognize the Covid LFTs, interpret their results, locate the correct visual evidence needed to interpret the test results, and detect partially obscured LFT tests.

We publicly-share our dataset to facilitate future progress on this challenging problem. Success can benefit other related applications, including automated analysis of other LFT test results including for pregnancy, HIV, and malaria. Our work also contributes to designing more interpretable solutions, by enabling assessment of the extent to which models reason based on the appropriate visual evidence.

2. Related Works

Lateral Flow Tests (LFTs). LFTs have been widely adopted for decentralized testing due to their simplicity, cost-effectiveness, and quick results. They are used for detecting a broad range of health conditions including pregnancy, Covid, and infectious diseases such as malaria and HIV. Each lateral flow test works by quantifying the presence of a target substance in a liquid sample (e.g., urine, saliva) to determine the presence of a medical condition. Each test achieves this by transporting the sample through pads where nanoparticles with specialized receptors react with the target substance that would indicate a positive result. The result is a visual signal indicating the strength of that target substance's presence. Our work will complement the existing work on automatically interpreting LFT results [11, 21, 31], by offering a new labeled dataset to support richer evaluation alongside the first published analysis of foundation models' zero-shot performance in interpreting LFT test results.

LFT Datasets. A limited number of large-scale, publicly available LFT image datasets exist [4, 29]. Our work is the first to enrich such images with annotations indicating the test result as well as the location of the test and its nested results window. This new dataset can empower researchers to improve the precision of automated visual inspections for LFT images by enabling them to verify models look at the correct visual evidence when making its predictions regarding the test results.

Vision Language Models. Many large VLMs [6, 8, 14, 15, 23, 30, 32] have made significant strides in various vision and cross-modality downstream tasks. For example, several models [15, 30] have shown strong performance on conventional public image captioning datasets like COCO-Captions [12], TextCaps [24], NoCaps [3], and Flickr30k [22], respectively. Such models often achieve their impressive reasoning and generalization capabilities by aligning visual features extracted from images with the input embedding space of power large language models (LLMs) like ChatGPT [18], GPT-4 [1], Vicuna [7], and LLaMA [28]. A well-known challenge from such models is the tendency to hallucinate, ignoring the image and instead specifying text the LLM knows should often appear together from training. We rigorously explore modern VLMs to interpret LFT images using specific prompts designed to direct them to locate the LFT tests and the test result window within the images, we aimed to assess the models' ability to draw interpretations from visual cues. Our findings shed light on the current capabilities of VLMs in accurate LFT image analysis, highlighting potential areas of improvement, as it directly impacts the reliability and effectiveness of accurate and unbiased healthcare image interpretation.

3. LFT-Grounding Dataset

We now introduce our extended version of LFT image dataset [17], that we call LFT-Grounding to reflect the dataset provide **g**roundings that localize LFT tests and their nested test result windows.

3.1. Dataset Creation

Source. We extend an existing dataset of 325 Covid Lateral Flow Test (LFT) images acquired in real-world settings, licensed by MIT on Kaggle [17]. Our dataset is restricted to these images due to obstacles in acquiring additional publicly-available images of LFTs.[1] Images are categorized according to their ground-truth labels, distinguishing between positive and negative results for LFT tests. All images contain exactly one clearly-visible, valid LFT test with observable test lines in its result window.

Annotation Task Design. We created an annotation interface for segmenting parts of LFT images. It starts with detailed instructions at the top that encompass navigating the interface and completing annotations for images with both positive and negative results, including an annotated example of each scenario. Then, annotators are walked through annotating five LFT images. For each image, users are first asked, "Is the Covid test positive? Please indicate "Yes" or "No" for the image you are viewing." After collecting the result of the Covid Test, annotators are tasked with outline the entire Covid LFT test within the image. Upon successful demarcation, annotators are then directed to segment the result window. Annotators are first asked, "Can you locate

---

[1] Other known images are HIV LFT images in [29] and pregnancy LFT images from Adobe Stock [2]. However, we are still waiting on our application on February 15,2024 for the former dataset to be approved and the latter restricts their redistribution in the license.

the Covid Test Result on the Covid Test Object?" and, if yes, are then instructed to delineate the result window. The interface enables segmentation by gathering a sequence of clicked points to form a contiguous polygon.

Annotation Collection. We hired crowdworkers from Amazon Mechanical Turk (AMT) to annotate the data. We deployed a total of 65 HITs to annotate all 325 images. We engaged 16 highly skilled annotators from AMT for this task, who already had contributed to previous efforts from our team in the preceding 30 months for large-scale data segmentation. The authors visually reviewed all submissions to verify they all are high-quality.

3.2. Dataset Analysis

We now characterize our LFT-Grounding dataset with respect to it's overall composition as well as spatial statistics characterizing the Covid tests and their result windows.

Overall Dataset Composition. In Table 1, we report for LFT-Grounding the total number of images, number of images showing a positive test result, number of images showing a negative test result, and total number of collected segmentations (recall, each image contains two segmentations with one for the test and one for the result window). Overall, we observe a large dataset imbalance with only 8% of images indicating a negative result. We suspect this could be attributed in part to people preferring to post images of positive Covid test results online in order to either warn their community or garner emotional support.

Covid LFT Test Statistics. We next characterize the segmentations for the test and result window with respect to four metrics:

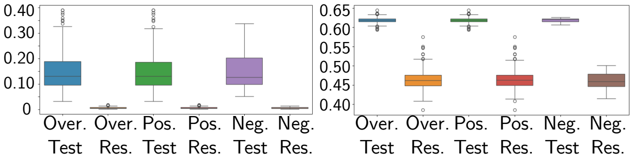
- **Image Coverage:** fraction of image pixels covered by a segmentation.
- **Test Coverage:** fraction of test pixels covered by the result window.
- **Boundary Complexity (BC):** ratio of a segmentations area to the length of it's perimeter (i.e., the isoperimetric quotient). Values are in $[0, 1]$ with 0 representing a highly jagged boundary and 1 representing a perfect circle.
- **Normalized Aspect Ratio (NAR):** Ratio of the shortest side of the segmentation to it's longest side. Values are in $(0, 1]$, where values approaching 0 represent a thin segmentation, while 1 represents a perfect square.

For all four metrics, we visualize the distribution of scores in Figure 2.

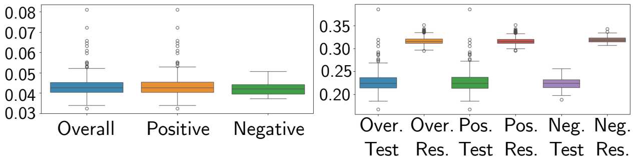
Overall, results for positive and negative results look similar across all metrics. The one exception is for outliers in positive tests, which we suspect is due to sample imbalance between positive and negative images in our dataset.

# Images	**# Pos.**	**# Neg.**	**# Seg.**
325	300	25	650

Table 1. Composition of LFT-Grounding. From left to right: number of images, number of images showing a positive Covid result, number of images showing a negative Covid result, and total number of segmentations.



(a) Image Coverage (b) Boundary Complexity



(c) Test Coverage (d) NAR

Figure 2. Boxplots for each of the four metrics used to analyze LFT-Grounding. Each boxplot shows the overall results, alongside fine-grained result for positive and negative tests. (a) image coverage; (b) boundary complexity; (c) test coverage; (d) NAR. The lines in each boxplot represent medians, the bottoms and tops of each boxplot represent the 25th and 75th percentiles respectively, whiskers represent most extreme data not considered outliers, and circles represent outliers. (Res.=Result window; Over.=Overall; Pos.=Positive; Neg.=Negative; NAR=Normalized aspect ratio)

We also observe that the result window occupies a small proportion of pixels in the images as well as within the Covid tests. This highlights a challenge for modern models [27], as locating small regions is known to be a challenging problem. Improving automated analysis of such small result windows would be especially valuable for visually impaired users, as the design of result windows is an obstacle for many low-vision users who have some lingering sight yet cannot independently discern information in such small physical regions.

When observing boundary complexity, we observe the the test itself is more circular than the result window (i.e., has a higher BC score). Intuitively, Covid tests typically have rounded corners, which contributes to having a simpler shape, whereas test windows are typically rectangular with sharp edges, leading to a more jagged shape, as demonstrated in Figure 1. Regions with simple boundaries (e.g., rectangles) may be easier for models to segment due to clear boundaries and less variability in form, compared to more complex shapes (e.g., humans). Finally, we observe that both the result window and rectangular shapes, with Covid tests having a NAR range in $[0.29, 0.35]$ (mean 0.32) and result windows having a range in $[0.17, 0.38]$ (mean 0.23), indicating that result windows are on average more narrow than the test itself, matching qualitative samples from our LFT-Grounding dataset in Figure 1. As a consequence, models may have trouble accurately localizing result windows as thin objects have shown to be challenging for modern models [13, 16].

4. Algorithm Benchmarking

We next analyze the performance of modern VLMs at analyzing images in our LFT-Grounding dataset.

Models. We chose to benchmark the following eight off-the shelf vision language models in zero-shot settings: BLIP2 [14], InstructBLIP [8], MiniGPT-4 [32], CogVLM [30], Monkey [15], GLaMM [23], ViP-LLaVA [6] and GPT-4V [1].[2] For InstructBLIP and MiniGPT-4, we test both available backbone options. We summarize these models in Table 2.

Evaluation Metrics. We use two evaluation metrics to assess a model's abilities to recognize the image contents and to do so based on the correct visual evidence.

For assessing recognition, we measure **accuracy** based on each model's ability to both identify that the test is for Covid and provide the correct test result. We establish a single score per image through two sequential steps of string matching. First, we search for "Covid-19 Test" (case-insensitive) in each model's generated caption. If that string is detected, then we perform string matching for *"positive result"* and *"negative result"* to find matches to the ground truth labels.[3] We penalize models as predicting wrong when generated captions include "positive or negative".[4]

Model	**Visual Input**	**Grounding Output**	**Grounding Type**
BLIP2 [14]	VP	×	–
InstructBLIP(Vicuna) [8]	VP	×	–
InstructBLIP(FlanT5) [8]	VP	×	–
MiniGPT-4(Vicuna) [32]	VP	×	–
MiniGPT-4(Llama) [32]	VP	×	–
GPT-4V [1]	VP	×	–
CogVLM [30]	VP,Coor	✓	*Bbox*
Monkey [15]	VP	×	–
GLaMM [23]	VP,Coor	✓	*Mask*
ViP-Llava [6]	VP,ROI	×	–

Table 2. Comparison of different methods in terms of input visual prompt formats and ability to generate groundings as visual output. Input formats include a text prompt along with an image (**VP**), text prompt specifying image coordinates along with an image (**Coor**), text prompt along with a region of interest indicated by visual cues such as arrows, bounding boxes, circles, scribbles overlaid onto the input image (**ROI**). Grounding output formats are bounding box (**Bbox**) and binary segmentation mask (**Mask**).

For models that can generate visual groundings (i.e., CogVLM and GLaMM) (Table 2), we evaluate their grounding capabilities by measuring the **Intersection over Union (IoU)** between the prediction and ground-truth. We do this separately for each entity type of interest, specifically the covid test and its test result window. Due to the different grounding outputs from these models (Table 2), we compare bounding boxes for CogVLM and segmentation masks for GLaMM.

For both measures, we compute the scores across all images. Then, we present results as percentages, where higher values range from 0 to 100 with higher scores signifying better performance.

Test Result Recognition with General Prompts. We first test three prompt formats: P_{G1}: "*Describe the image in detail*", P_{G2}: "*Describe in detail every object and their parts*", and P_{G3}: "*Describe the hierarchical parts of the LFT test in the image*". Results are shown in Table 3.

While most models perform poorly at recognizing the test result, we observe strong performance from GPT-4V with scores ranging from 64% to 74% across the three prompts. We suspect GPT-4V's strong performance is due to its more extensive training data, although its proprietary design limits further analysis.

When comparing the performance of different prompts, we observe marginally improved accuracy scores for the prompt P_{G1} compared to prompts P_{G2} and P_{G3}. We suspect this is due to the prompt's closer resemblance to those used during the training of these models; e.g., *"Describe this image in detail"* was used when fine-tuning MiniGPT-4 [32] and *"Generate the detailed caption in English:"* was used when instruction tuning Monkey [15]).

Test Result Recognition With Prompts Specifying the Test Type. We next prompt the model with additional information about the test type as follows: "P_{S1}: *Describe in detail every part of Covid test in the image*", "P_{S2}: *Describe the hierarchical parts of Covid test in the image*". Results are reported in the Table 3. We observe an average improvement of $\sim$5% in accuracy scores when we notify models about the test type as opposed to general prompts. We suspect specifying the test type helps models know that common language patterns are to specify a test result when reporting the presence of a Covid test.

---

[2] While the commercial application, Be My AI [20], has potential for reading LFT's results, it currently only exists as a mobile application and so is unsuitable for our large-scale evaluation.

[3] Preliminary findings showed better evaluation using "result" in the string rather than simply searching for "positive" and "negative".

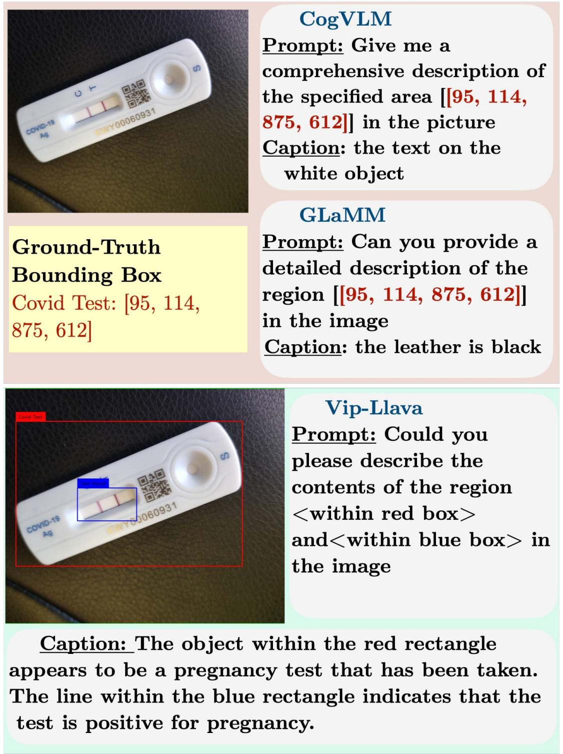
[4] Preliminary findings gave similar outcomes when instead searching for two strings: "positive" and "negative".

Model	General Prompt			Specific Prompt	
	$\mathbf{P}_{\mathbf{G1}}$	$\mathbf{P}_{\mathbf{G2}}$	$\mathbf{P}_{\mathbf{G3}}$	$\mathbf{P}_{\mathbf{S1}}$	$\mathbf{P}_{\mathbf{S2}}$
BLIP2 [14]	0	0	0	0	0
InstructBLIP(Vicuna) [8]	0	0	0	0.62	0.92
InstructBLIP(FlanT5) [8]	0	0	0	0	0
MiniGPT-4(Vicuna) [32]	0.31	0.31	1.23	4.92	2.77
MiniGPT-4(Llama) [32]	0.31	0	0	3.08	2.77
GPT-4V [1]	74.15	64.62	67.38	62.24	74.77
Monkey [15]	2.77	0.62	0	0.62	0
CogVLM [30]	23.69	15.08	4.62	30.77	7.38
GLaMM [23]	0	0	0	0	0
ViP-Llava [6]	3.38	2.77	0	23.69	8.62

Table 3. Test result recognition performance from models prompted in a zero-shot setting. $P_{G/S}$ refers to different general and specific prompts respectively, as described in (Section 4).

Test Result Recognition When Notifying Model's About the Covid Test's Location. We next assess the models' perfomance when they are notified where to look in the given image. We feed bounding box coordinates of the Covid test's location (derived from its ground-truth segmentation) to the models that accept coordinates as inputs to explore the upper bound of what these models can achieve. We use predefined prompts for visual inputs from the original documentations of CogVLM [26], GLaMM [23], and VipLlava [6].[5] Quantitative results are reported in Table 4 and a qualitative result is shown in Figure 3.

We find that all models generally perform worse on our dataset when given ground-truth bounding box coordinates through prompts. While Vip-Llava misinterprets the Covid LFT image as 'pregnancy test' in 99% of the cases, as exemplified in the Figure 3, it slightly surpasses its counterparts (i.e., CogVLM and GLaMM). We believe Vip-Llava's improved performance stems from its inherent model design [6]. Unlike CogVLM and GLaMM, which are limited to understanding information conveyed through textual prompts, Vip-Llava is uniquely designed to interpret both the overlaid visual markers (specifically, bounding boxes that identify Covid Test and Test Result Window in our case) on the image and the accompanying prompt. This dual-layered approach enables Vip-Llava to generate better descriptions for the LFT images.



CogVLM

Prompt: Give me a comprehensive description of the specified area [[95, 114, 875, 612]] in the picture

Caption: the text on the white object

Ground-Truth Bounding Box

Covid Test: [95, 114, 875, 612]

GLaMM

Prompt: Can you provide a detailed description of the region [[95, 114, 875, 612]] in the image

Caption: the leather is black

Vip-Llava

Prompt: Could you please describe the contents of the region <within red box> and<within blue box> in the image

Caption: The object within the red rectangle appears to be a pregnancy test that has been taken. The line within the blue rectangle indicates that the test is positive for pregnancy.

Figure 3. Example of generated captions by the models when they are notified in the prompts of the Covid test's location.

Evaluating VLMs with Grounding Abilities in Zero-shot Setting for LFT Recognition We assess the visual grounding proficiency of two models supporting this capability: CogVLM and GLaMM. For CogVLM, which supports bounding box predictions, we design the prompt: *"Please describe the <entity >in detail and provide its coordinates [[x0, y0, x1, y1]]"*. For GLaMM, which supports

---

[5] For CogVLM, we use the prompt template: "*Give me a comprehensive description of the specified area [[x0,y0,x1,y1]] in the picture*". For GLaMM, we use: "*Can you provide a detailed description of the region [[x0,y0,x1,y1]] in the image*". Unlike CogVLM and GLaMM, which accept bounding boxes through text prompts, Vip-Llava processes the image with overlaid multiple bounding boxes, accompanied by a text prompt specifying the visual cues (e.g., " <red bounding box >", "green circle"). The prompt used for Vip-Llava is: *"Could you please describe the contents of the region <within red box >and <within blue box >in the image"* (where <within red box >refers to the Covid LFT Test object's bounding box and <within blue box >refers to the LFT Test Result Window).

Model	**Bbox Prompt**
CogVLM [30]	0
GLaMM [23]	0
ViP-Llava [6]	0.31

Table 4. Accuracy metrics for correctly indicating positive and negative test results in generated captions with respect to Covid LFT images in the dataset, when models operate in their zero-shot setting using auxiliary prompts which provide ground-truth bounding box coordinates for the Covid test.

Model	**IoU**$_{LFT-Test}$	**IoU**$_{Result-Window}$
CogVLM [30]	28	16.09
GLaMM [23]	97.59	4.69

Table 5. Overall performance of CogVLM [30] and GLaMM [23] in locating the Covid LFT test as well as its nested result window for all images in our dataset (mean value reported).

segmentation predictions, we design the prompt: *"Can you please segment <entity >in the given image"*. We use for <entity >both "Covid Test" and "Covid Test Result Window". Results are shown in Table 5.[6]

For locating the Covid tests, we observe higher IoU scores from GLaMM than CogVLM. We suspect superior performance of GLaMM stems in part from providing fine-grained pixel-level object groundings rather than coarse bounding boxes like CogVLM. Additionally, we attribute this to the observation that in 7% of cases when CogVLM is prompted for object "Covid Test", and in $< 2\%$ of cases when prompted for "Test Result Display Window", it fails to provide the grounding coordinates for the specified salient object in the prompt. A commonality shared in these cases is either the LFT test is partially obscured (e.g., from being held in a hand) or the LFT test is placed on a dark or highly textured background, as exemplified in Figure 4.

When observing performance gaps between average IoU scores for the predicted bounding box for the Covid test and test result window, we see that both models struggle to identify the nested test result window within the LFT images. Specifically, the performance decrease is $\sim$42.50% for CogVLM and $\sim$95.31% for GLaMM. We suspect this decline in performance is due to VLMs' limited grasp in recognizing hierarchical decomposition within objects as well as in interpreting small and thin entities. This finding underscores a valuable future direction in improving the visual grounding capabilities at multiple decomposition levels that include smaller entities.

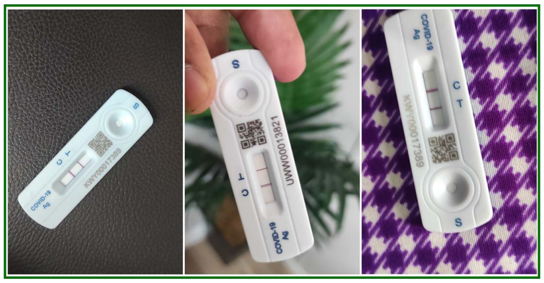


Figure 4. Examples when CogVLM did not generate bounding box predictions for both the Covid test and its result window.

Qualitative Results We display the captions generated by the models for the top-performing prompt *"Describe the image in detail"* in Figure 6.

Upon visual inspection, we observe a common mistake from models is misidentifying the Covid LFT image as a "pregnancy test," "thermometer," or "blood glucose monitor." While these errors may stem from model hallucinations, we also suspect this poor performance is due to training datasets lacking sufficient diversity of LFT test images.

We also show an example of the predictions for the Covid test and the nested test result window by GLaMM in Figure 5. This reinforces our quantitative finding that performance declines between from the Covid test to the locating its nested result window.

5. Conclusions

We introduce LFT-Grounding to catalyze research into improving the zero-shot generalization capabilities of state-of-the-art models for automatic interpretation of LFT images. We also benchmark eight state-of-the-art models in zero-shot setting to highlight their current status for this challenging problem. Our work underscores opportunities for future work, including on resolving how to acquire LFT images for a wider range of health conditions as well as on improving the performance of automatic models in interpreting results and achieving this by conveying the visual evidence it used to arrive at that interpretation. An important step for future work will include expanding our test result categorizations to also support recognizing "invalid" test results. We publicly share our dataset to spur community effort to facilitate future extensions of this work.

Acknowledgments. This project was supported in part by a National Science Foundation SaTC award (#2148080) and gift funding from Microsoft AI4A. Josh Myers-Dean is supported by a NSF GRFP fellowship (#1917573). We thank Bo Xie, Tom Yeh, and Dan Larremore for helping inspire this research direction.

---

[6] We exclude performance scores for recognizing the test result because in most instances the models didn't output a string description of the image.

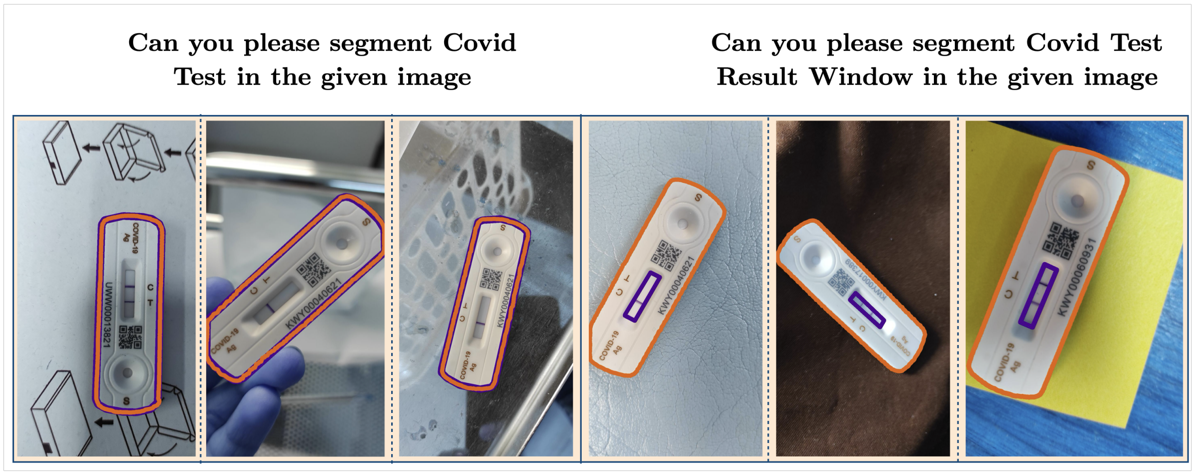
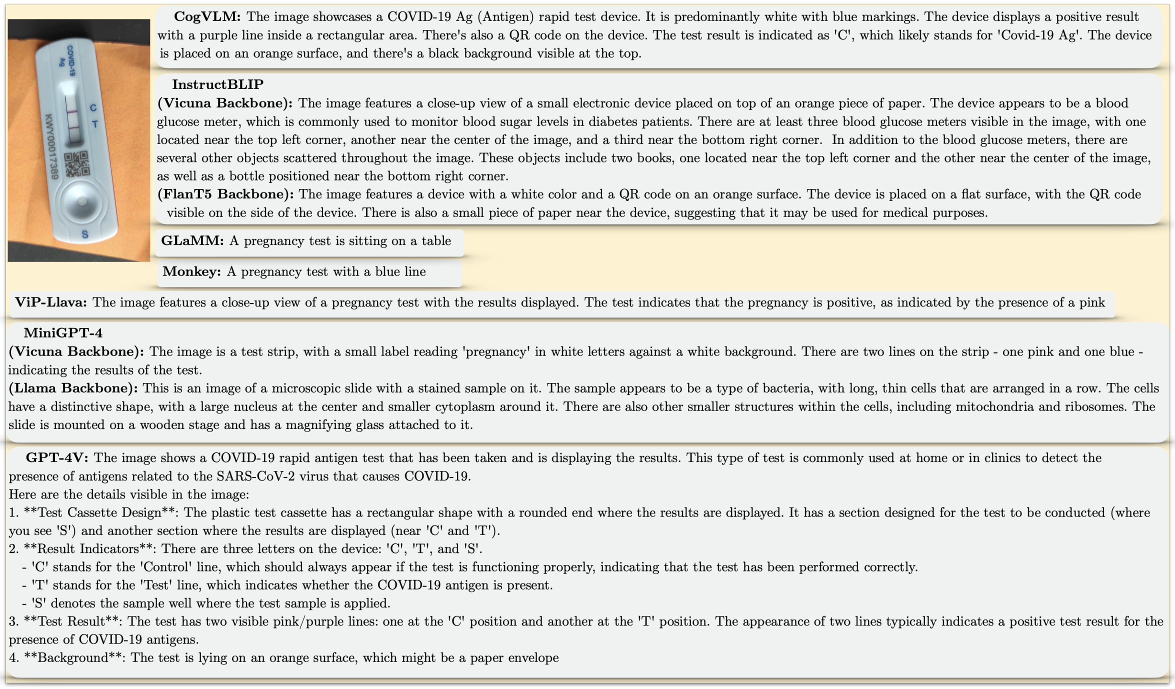

Can you please segment Covid Test in the given image
Can you please segment Covid Test Result Window in the given image

Figure 5. Examples of ground-truth (purple overlay) and GLaMM predictions (orange overlay) for locating the Covid test as well as its nested result window.



CogVLM: The image showcases a COVID-19 Ag (Antigen) rapid test device. It is predominantly white with blue markings. The device displays a positive result with a purple line inside a rectangular area. There's also a QR code on the device. The test result is indicated as 'C', which likely stands for 'Covid-19 Ag'. The device is placed on an orange surface, and there's a black background visible at the top.

InstructBLIP
(Vicuna Backbone): The image features a close-up view of a small electronic device placed on top of an orange piece of paper. The device appears to be a blood glucose meter, which is commonly used to monitor blood sugar levels in diabetes patients. There are at least three blood glucose meters visible in the image, with one located near the top left corner, another near the center of the image, and a third near the bottom right corner. In addition to the blood glucose meters, there are several other objects scattered throughout the image. These objects include two books, one located near the top left corner and the other near the center of the image, as well as a bottle positioned near the bottom right corner.
(FlanT5 Backbone): The image features a device with a white color and a QR code on an orange surface. The device is placed on a flat surface, with the QR code visible on the side of the device. There is also a small piece of paper near the device, suggesting that it may be used for medical purposes.

GLaMM: A pregnancy test is sitting on a table

Monkey: A pregnancy test with a blue line

ViP-Llava: The image features a close-up view of a pregnancy test with the results displayed. The test indicates that the pregnancy is positive, as indicated by the presence of a pink

MiniGPT-4
(Vicuna Backbone): The image is a test strip, with a small label reading 'pregnancy' in white letters against a white background. There are two lines on the strip - one pink and one blue - indicating the results of the test.
(Llama Backbone): This is an image of a microscopic slide with a stained sample on it. The sample appears to be a type of bacteria, with long, thin cells that are arranged in a row. The cells have a distinctive shape, with a large nucleus at the center and smaller cytoplasm around it. There are also other smaller structures within the cells, including mitochondria and ribosomes. The slide is mounted on a wooden stage and has a magnifying glass attached to it.

GPT-4V: The image shows a COVID-19 rapid antigen test that has been taken and is displaying the results. This type of test is commonly used at home or in clinics to detect the presence of antigens related to the SARS-CoV-2 virus that causes COVID-19.
Here are the details visible in the image:
1. **Test Cassette Design**: The plastic test cassette has a rectangular shape with a rounded end where the results are displayed. It has a section designed for the test to be conducted (where you see 'S') and another section where the results are displayed (near 'C' and 'T').
2. **Result Indicators**: There are three letters on the device: 'C', 'T', and 'S'.
   - 'C' stands for the 'Control' line, which should always appear if the test is functioning properly, indicating that the test has been performed correctly.
   - 'T' stands for the 'Test' line, which indicates whether the COVID-19 antigen is present.
   - 'S' denotes the sample well where the test sample is applied.
3. **Test Result**: The test has two visible pink/purple lines: one at the 'C' position and another at the 'T' position. The appearance of two lines typically indicates a positive test result for the presence of COVID-19 antigens.
4. **Background**: The test is lying on an orange surface, which might be a paper envelope

Figure 6. Models' predictions using the overall top-performing prompt, *"Describe the image in detail"*, for a test with a positive result.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 2, 4, 5

[2] Adobe. Stock photos, royalty-free images, graphics, vectors videos. 2

[3] Harsh Agrawal, Karan Desai, Yufei Wang, Xinlei Chen, Rishabh Jain, Mark Johnson, Dhruv Batra, Devi Parikh, Stefan Lee, and Peter Anderson. Nocaps: Novel object captioning at scale. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8948–8957, 2019. 2

[4] Robert Banathy, Mark Branigan, Paul Lewis-Borman, Nishali Patel, Lennard Lee, Tom Alan Fowler, Camila Caiado, Anna Dijkstra, Piotr Chudzik, Paria Yousefi, et al. Machine learning for determining lateral flow device results in asymptomatic population: a diagnostic accuracy study. 2021. 1, 2

[5] Jobie Budd, Benjamin S Miller, Nicole E Weckman, Dounia Cherkaoui, Da Huang, Alyssa Thomas Decruz, Noah Fongwen, Gyeo-Re Han, Marta Broto, Claudia S Estcourt, et al. Lateral flow test engineering and lessons learned from covid-

19. *Nature Reviews Bioengineering*, 1(1):13–31, 2023. 1

[6] Mu Cai, Haotian Liu, Siva Karthik Mustikovela, Gregory P. Meyer, Yuning Chai, Dennis Park, and Yong Jae Lee. Making large multimodal models understand arbitrary visual prompts. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2024. 2, 4, 5, 6

[7] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*, 2023. 2

[8] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning, 2023. 2, 4, 5

[9] Dcard. White house responds to ndrn's concerns about covid-19 test accessibility, Jan 2022. 1

[10] Noah A. Guillermo, Rohan Chawla, Elaine M. Zheng, Adel A. Battikha, Kyle J. Chen, and Daniel T. Kamei. Development of a universal lateral-flow immunoassay interpretation device prototype with quantitative output for blind or visually impaired individuals. *IEEE Sensors Journal*, 23(19):23536–23544, 2023. 1

[11] Sjoukje-Marije Haisma, Anne Galaurchi, Shatha Almahwzi, Joy A Adekanmi Balogun, Anneke C Muller Kobold, and Patrick F van Rheenen. Head-to-head comparison of three stool calprotectin tests for home use. *PLoS One*, 14(4):e0214751, 2019. 1, 2

[12] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3128–3137, 2015. 2

[13] Lei Ke, Mingqiao Ye, Martin Danelljan, Yu-Wing Tai, Chi-Keung Tang, Fisher Yu, et al. Segment anything in high quality. *Advances in Neural Information Processing Systems*, 36, 2024. 4

[14] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023. 2, 4, 5

[15] Zhang Li, Biao Yang, Qiang Liu, Zhiyin Ma, Shuo Zhang, Jingxu Yang, Yabo Sun, Yuliang Liu, and Xiang Bai. Monkey: Image resolution and text label are important things for large multi-modal models. *arXiv preprint arXiv:2311.06607*, 2023. 2, 4, 5

[16] Jun Hao Liew, Scott Cohen, Brian Price, Long Mai, and Jiashi Feng. Deep interactive thin object selection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 305–314, 2021. 4

[17] Mahdi. Covid19 lateral flow test images, Dec 2023. 1, 2

[18] OpenAI. Chatgpt, 2023. 2

[19] Sushil K Oswal and Hitender K Oswal. Study of a smartphone app as a bridge assistive technology for a covid-19 home test: 19 essential guidelines. In *Cambridge Workshop on Universal Access and Assistive Technology*, pages 118–126. Springer, 2023. 1

[20] Michele Paris. Introducing: Be my ai. 4

[21] Chunjong Park, Hung Ngo, Libby Rose Lavitt, Vincent Karuri, Shiven Bhatt, Peter Lubell-Doughtie, Anuraj H Shankar, Leonard Ndwiga, Victor Osoti, Juliana K Wambua, et al. The design and evaluation of a mobile system for rapid diagnostic test interpretation. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 5(1):1–26, 2021. 1, 2

[22] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer vision*, pages 2641–2649, 2015. 2

[23] Hanoona Rasheed, Muhammad Maaz, Sahal Shaji, Abdelrahman Shaker, Salman Khan, Hisham Cholakkal, Rao M. Anwer, Eric Xing, Ming-Hsuan Yang, and Fahad S. Khan. Glamm: Pixel grounding large multimodal model. *ArXiv 2311.03356*, 2023. 2, 4, 5, 6

[24] Oleksii Sidorov, Ronghang Hu, Marcus Rohrbach, and Amanpreet Singh. Textcaps: a dataset for image captioning with reading comprehension. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 742–758. Springer, 2020. 2

[25] Kelley Smith. Group continues to fight for accessible at-home covid-19 tests for the blind, Feb 2022. 1

[26] Thudm. Thudm/cogvlm: A state-of-the-art-level open visual language model: . 5

[27] Kang Tong and Yiquan Wu. Deep learning-based detection from the perspective of small or tiny objects: A survey. *Image and Vision Computing*, 123:104471, 2022. 3

[28] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 2

[29] Valerian Turbe, Carina Herbst, Thobeka Mngomezulu, Sepehr Meshkinfamfard, Nondumiso Dlamini, Thembani Mhlongo, Theresa Smit, Valeriia Cherepanova, Koki Shimada, Jobie Budd, et al. Deep learning of hiv field-based rapid tests. *Nature medicine*, 27(7):1165–1170, 2021. 1, 2

[30] Weihan Wang, Qingsong Lv, Wenmeng Yu, Wenyi Hong, Ji Qi, Yan Wang, Junhui Ji, Zhuoyi Yang, Lei Zhao, Xixuan Song, et al. Cogvlm: Visual expert for pretrained language models. *arXiv preprint arXiv:2311.03079*, 2023. 2, 4, 5, 6

[31] Nathan CK Wong, Sepehr Meshkinfamfard, Valérian Turbé, Matthew Whitaker, Maya Moshe, Alessia Bardanzellu, Tianhong Dai, Eduardo Pignatelli, Wendy Barclay, Ara Darzi, et al. Machine learning to support visual auditing of home-based lateral flow immunoassay self-test results for sars-cov-2 antibodies. *Communications medicine*, 2(1):78, 2022. 1, 2

[32] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023. 2, 4, 5