MLLMs KNOW WHERE TO LOOK: TRAINING-FREE PERCEPTION OF SMALL VISUAL DETAILS WITH MULTIMODAL LLMS

Jiarui Zhang, Mahyar Khayatkhoei, Prateek Chhikara, Filip Ilievski

University of Southern California, USA
Vrije Universiteit Amsterdam, The Netherlands

ABSTRACT

Multimodal Large Language Models (MLLMs) have experienced rapid progress in visual recognition tasks in recent years. Given their potential integration into many critical applications, it is important to understand the limitations of their visual perception. In this work, we study whether MLLMs can perceive small visual details as effectively as large ones when answering questions about images. We observe that their performance is very sensitive to the size of the visual subject of the question, and further show that this effect is in fact causal by conducting an intervention study. Next, we study the attention patterns of MLLMs when answering visual questions, and intriguingly find that they consistently know where to look, even when they provide the wrong answer. Based on these findings, we then propose training-free visual intervention methods that leverage the internal knowledge of any MLLM itself, in the form of attention and gradient maps, to enhance its perception of small visual details. We evaluate our proposed methods on two widely-used MLLMs and seven visual question answering benchmarks and show that they can significantly improve MLLMs' accuracy *without requiring any training*. Our results elucidate the risk of applying MLLMs to visual recognition tasks concerning small details and indicate that visual intervention using the model's internal state is a promising direction to mitigate this risk.[1]

1 INTRODUCTION

Multimodal large language models (MLLMs) (Hurst et al., 2024; Team et al., 2024; Anthropic, 2024; Wang et al., 2024; Li et al., 2024a; Team et al., 2025; Chen et al., 2025) have greatly progressed the state of multimodal reasoning and planning, and are rapidly being integrated into various downstream applications, ranging from robotics (Li et al., 2024b; Chen et al., 2024), biomedicine (Li et al., 2023a), autonomous driving (Xu et al., 2024b; Zhang et al., 2023a) to visual mathematical reasoning (Gao et al., 2023; Zhang et al., 2024c;b) and even food recipe generation (Chhikara et al., 2024). Given the rapidly growing application of MLLMs, especially in critical domains such as biomedicine and security, it is crucial to study the limitations of their visual perception to elucidate the potential risks that may affect their downstream applications.

To motivate the limitation that will be the focus of this work, we start by presenting three revealing visual question answering examples in Fig. 1, in which we ask a popular MLLM BLIP-2 (FlanT5$_{\text{XL}}$) (Li et al., 2023b) to identify an object's presence or type in each image as we vary the size of the object. In the absence of any prior evidence, we might reasonably expect the MLLM's answer to be invariant to the size of the object, because of the MLLM's large representational capacity and pretraining on a wide variety of images containing objects of various sizes. To the contrary, in Fig. 1 (left), we observe that initially the model does not recognize the existence of a small street sign and assigns a lower probability to the correct answer; however, zooming into the image (via more focused visual cropping) towards the street sign gradually increases the probability assigned to the correct answer, suggesting that the model gradually perceives more and more relevant details of the street sign.

[1] Our code is available at https://github.com/saccharomycetes/mllms_know.

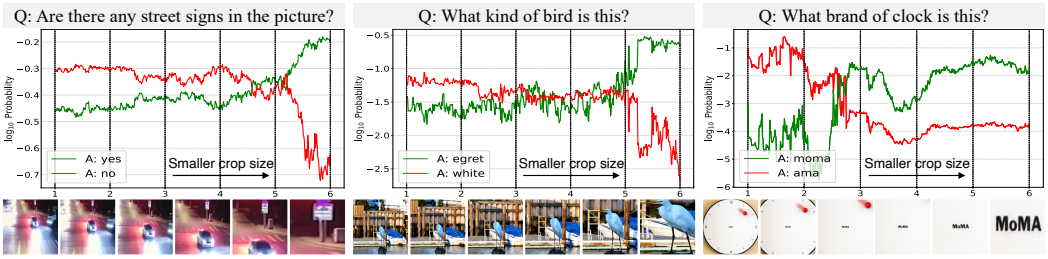

Figure 1: The effect of visual cropping on the probability of answers predicted by BLIP-2 FlanT5$_{\text{XL}}$ zero-shot VQA model. The x-axis labels are indices to the respective cropped images displayed under each plot that the model sees at each step. The model gradually finds the correct answer.

In Fig. 1 (middle), we observe further evidence of this difficulty in perceiving small details: the model initially predicts *white* as the type of the bird, but when we zoom into the image towards the bird, without changing the question in any way, we observe that the model gradually assigns higher probability to the correct bird type of *egret*. This suggests that the model was not making a semantic error of misunderstanding what *type* means, rather it was unable to perceive sufficient details to discriminate *egret* from other *white* birds, which is mitigated by visual cropping. Similarly, in Fig. 1 (right), we observe that the model's initial answer is not entirely irrelevant ("ama" compared to the correct answer "moma"), suggesting that the model knows where to look based on the question but cannot accurately perceive the actual word, which is again mitigated by visual cropping.

In this work, we will study the limitation observed in Fig. 1 extensively, elucidate its cause, and propose potential solutions to mitigate its consequences. In Sec. 3, we quantitatively show that there indeed exists a difficulty in perceiving small visual concepts across various widely-used MLLMs. Our findings are consistent with prior works on evaluating the text-image matching in vision-language joint embedding models, which have observed a reverse correlation between visual object size in images and the text-image matching score (Zhao et al., 2022), but we further establish a causal connection between visual concept size and MLLMs' perception ability through an intervention study. In Sec. 4, we study whether the MLLMs' difficulty in perceiving small visual concepts stems from a difficulty in perceiving visual details, or from a difficulty in locating the concept due to its small size. We quantitatively show that MLLMs consistently know where to look, even when they fail to answer the question correctly. In Sec. 5, we propose three automatic visual cropping methods—leveraging the attention maps and gradients of the MLLM itself—as scalable and training-free solutions to the visual perception limitation. Finally, in Sec. 6, we apply our proposed methods to two popular MLLMs and evaluate them on seven visual question answering (VQA) benchmarks, showing their efficacy in enhancing MLLMs' accuracy, especially on detail-sensitive benchmarks.

2 Related Works

Multimodal Large Language Models (MLLMs). MLLMs are foundation models capable of handling diverse language and vision tasks. These models fall into two categories: *end-to-end pretrained* and *modular pretrained*. End-to-end models process joint image-language data through architectures such as dual-encoder (Radford et al., 2021), fusion-encoder (Li et al., 2021), encoder-decoder (Cho et al., 2021), and unified transformer (Wang et al., 2022), using objectives like image-text matching, contrastive learning, and masked language modeling. Modular pretrained models, which dominate recent state-of-the-art approaches, avoid costly full pretraining by adapting existing components: BLIP2 (Li et al., 2023b) and InstructBLIP (Dai et al., 2023) train a Transformer-based connector between a frozen pretrained ViT (Dosovitskiy et al., 2021) image encoder and a frozen LLM, which transforms ViT output tokens into a fixed set of image tokens in the input space of the LLM; Qwen-VL (Bai et al., 2023), similarly uses a fixed-length token connector (a single cross-attention layer), but trains both the connector and the LLM; LLaVA (Liu et al., 2023b) and LLaVA-1.5 (Liu et al., 2023a) instead use a linear projection and a two-layer MLP as their connectors, respectively, and train both. Our work will contribute to a better understanding of the perception limitations of MLLM and improve their perception scalably and without training, offering orthogonal benefits to existing approaches.

Visual Localization Methods. Dedicated visual localization methods, such as YOLO (Redmon et al., 2016), SAM (Kirillov et al., 2023), and GLIP (Li et al., 2022b), rely heavily on dense spatial annotations for identifying salient image regions. Native approaches, such as Grad-CAM (Selvaraju et al., 2017), localize regions by analyzing gradients from classifier decisions without spatial supervision. Prior work adapts Grad-CAM to BLIP (Li et al., 2022a) leveraging its dedicated image-text similarity computation neural network called the Image-Text Matching network (Tiong et al., 2022; Guo et al., 2023). In this work, we derived a more general way for localizing the attention of MLLMs to images, not relying on the specific BLIP architecture. Several recent works have explored ways to improve the visual localization capability of MLLMs for visual question answering, including chain-of-thought (Shao et al., 2024; Liu et al., 2024b), tool-using (Wu and Xie, 2023), and visual programming approaches (Surís et al., 2023; Gupta and Kembhavi, 2023). In contrast, we demonstrate that MLLMs can often effectively localize the visual subject of a question in their internal states, and propose training-free methods to leverage their internal states for improving their visual perception.

Visual Perception Limitations in MLLMs. The difficulty of answering questions about small objects in images has been observed by several prior and concurrent works (Zhang et al., 2023b; 2024a; Liu et al., 2024a; Wu and Xie, 2023), which have explored mitigating solutions based on high-resolution fine-tuning (Liu et al., 2024a; Dehghani et al., 2023; Wang et al., 2024), multi-agent pipelines (Wu and Xie, 2023), and use of visual cropping (Zhang et al., 2023b). In this work, we provide more extensive evidence for this difficulty, establish its causal effect on MLLMs' performance, and show that it is rooted in a failure to observe small visual details as opposed to a failure to locate small objects. Several works have also shown that MLLMs suffer from object hallucination (Li et al., 2023c; Yu et al., 2024). Furthermore, Zhang et al. (2024a) have shown visual blind spots in MLLMs—i.e., locations on the image where the MLLMs' perception degrades—as well as their sensitivity to visual quality, presence of visual distractors in the image, and even local object location perturbations.

3 MLLMs' Sensitivity to the Size of Visual Concepts

In this section, our goal is to quantitatively study our qualitative observations in Fig. 1 that MLLMs struggle with describing small visual details in images. To that end, we consider the TextVQA dataset, in which for each question we can find the image ground-truth bounding box that contains the correct textual answer. We partition its validation set into three groups based on the relative size of the ground-truth bounding box $S = \frac{A_{bb}}{A_{total}}$, where A_{bb} denotes the area of the ground-truth bounding box, and A_{total} the total area of the image: 1) $S < 0.005$ (`small`) consisting of 773 question-image pairs, 2) $0.005 \leq S < 0.05$ (`medium`) consisting of 2411 question-image pairs, and 3) $S \geq 0.05$ (`large`) consisting of 1186 question-image pairs. We chose TextVQA for this study because it contains

Table 1: Sensitivity of the accuracy of MLLMs to the size of visual concepts in TextVQA. As the relative visual size of the answer decreases (right to left in each row), we observe a decline in the accuracy of the original models (no cropping) in answering questions, whereas visual cropping (`human-CROP`) significantly improves accuracy on smaller objects.

Model	Method	Answer Bbox Size (S)		
		`small`	`medium`	`large`
BLIP-2 (FlanT5$_{\text{XL}}$)	no cropping	12.13	19.57	36.32
	`human-CROP`	55.76	52.02	45.73
InstructBLIP (Vicuna-7B)	no cropping	21.79	30.58	45.30
	`human-CROP`	69.60	61.56	53.39
LLaVA-1.5 (Vicuna-7B)	no cropping	39.38	47.74	50.65
	`human-CROP`	69.95	65.36	56.96
Qwen-VL (Qwen-7B)	no cropping	56.42	65.09	68.60
	`human-CROP`	70.35	75.49	71.05
GPT-4o	no cropping	65.76	72.81	69.17
	`human-CROP`	75.63	81.32	71.72

a significant number of questions about small visual concepts, and textual answers are harder for MLLMs to guess from other side information in the image (compared to object types and attributes).

Sensitivity Study. If a model's perception is not sensitive to the size of visual concepts, we expect it to have similar accuracy in all three partitions. In Tab. 1, we observe that the accuracy of all MLLMs declines as the ground-truth bounding box becomes relatively smaller (right to left on the *no cropping* rows). BLIP-2 and InstructBLIP are not trained on TextVQA (*i.e.*, are zero-shot models), and their accuracy declines by 24 and 23 absolute percentage points between the `large` and `small` partitions, respectively. LLaVA-1.5 and Qwen-VL are trained on the training set of TextVQA, yet, their accuracy also declines by 11 and 12 between the `large` and `small` partitions, respectively. Lastly, even the most recent commercial GPT-4o, with an unknown training set that might include all of TextVQA, is suffering from a 7 percentage point decline in accuracy between small and medium partitions. These findings suggest that MLLMs have a bias against perceiving smaller visual concepts.

Intervention Study. The perceptual limitation we observed above might be merely correlated with size. To study whether this limitation is causally related to size, we conduct an intervention study where we provide the MLLMs with visually cropped images based on the ground-truth bounding boxes, denoted as `human-CROP`. More specifically, for each image-question pair and each MLLM, we crop the smallest square-shaped region containing the ground-truth bounding box from the image, and resize it to the input image resolution of the MLLM (the square-shaped cropping prevents extreme deformation of the cropped image when resizing). The cropped image is then provided to the MLLM in addition to the original image-question pair (see more details in Fig. 4). We observe in Tab. 1 that `human-CROP` significantly improves the accuracy of all MLLMs on the `small` and `medium` partitions, and to a lesser extent on the `large` partition. These findings show that the perception limitation is indeed caused by the size of the visual concepts, and that visual cropping can be a promising direction to mitigate this limitation.

4 Do MLLMs Know Where to Look?

The limitation in perceiving small visual concepts can have two primary reasons: 1) they are hard to locate in the larger image, and 2) their small details are hard to perceive correctly. In Fig. 1, we observed that the MLLM's incorrect answer may contain partially correct information, hinting that it might know where to look in the image. In this section, we quantitatively study that observation to answer whether MLLMs' sensitivity to size is rooted in a perception limitation or a localization limitation. To that end, we first utilize the attention maps computed inside the Transformer layers of an MLLM to quantify its spatial attention over the image and then compare the total amount of this attention inside the ground-truth bounding box to other bounding boxes of the same size.

MLLMs' Setup. The considered MLLMs process a given image-question pair (x, q) in four steps (depicted in Fig. 4): 1) the image is divided into $N \times N$ non-overlapping patches and processed by the ViT image encoder into $N \times N$ output tokens; 2) the ViT output tokens are transformed into the input space of the backbone LLM—by either an MLP (LLaVA-1.5) or a Transformer connector (BLIP-2, InstructBLIP and Qwen-VL)—which we refer to as image tokens; 3) the image tokens are then prepended to the question tokens and a predefined starting answer token, and fed to the LLM; 4) the LLM is sampled auto-regressively following the starting answer token (we use greedy sampling).

Quantifying MLLMs' Spatial Attention over the Image. We first measure how important each image token is to the MLLM's decision (*answer-to-token attention*) by extracting the softmax cross-attention of the starting answer token to all image tokens in all layers of the backbone LLM, resulting in $A_{st}(x, q) \in \mathbb{R}^{L \times H \times 1 \times T}$, where L, H are the number of layers and heads-per-layer in the LLM, and T is the number of image tokens provided to the LLM. We then take the average over attention heads to arrive at the answer-to-token attention $\hat{A}_{st}(x, q) = \frac{1}{H} \sum_{h=1}^{H} A_{st}(x, q)$. Next, we measure how important each image region is to each image token (*token-to-image attention*). For the MLLMs that use a Transformer connector to resample ViT output tokens into a fixed number of image tokens (BLIP-2, InstructBLIP and Qwen-VL), we extract the softmax cross-attention of each image token to all ViT output tokens in all layers of the connector, resulting in $A_{ti} \in \mathbb{R}^{L_c \times H_c \times T \times N^2}$, where L_c, H_c are the number of layers and heads-per-layer in the connector, T the number of learnable query tokens (that become input image tokens to the LLM), and N^2 the number of image patches of the ViT image encoder. We then take the average over attention heads to arrive at the token-to-image attention

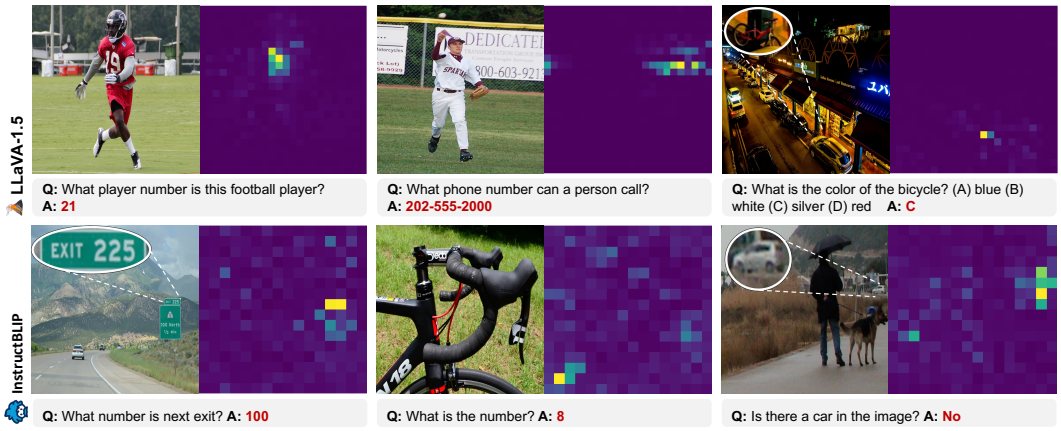

Figure 2: Examples of MLLMs knowing where to look despite answering incorrectly. The right panel in each example displays relative attention to the image (defined in Sec. 4) of one layer in the MLLM.

$\hat{A}_{ti}(x) = \frac{1}{H_c}\sum_{h=1}^{H_c} A_{ti}(x)$. For LLaVA-1.5 which uses an MLP to transform ViT output tokens to image tokens, we set $\hat{A}_{ti}(x)$ to the identity tensor. Finally, we compute the *answer-to-image attention* by computing the tensor product of the answer-to-token and token-to-image attention, resulting in $A_{si}(x,q) \in \mathbb{R}^{L\times L_c\times 1\times N^2}$ where $A_{si}^{mk}(x,q) = \hat{A}_{st}^{m}(x,q)\hat{A}_{ti}^{k}(x)$ (superscripts m and k denote layer indices on the LLM and the connector, respectively).

Relative Attention. One issue with using the softmax cross-attention is that not all highly attended tokens are semantically relevant to the input question. For example, recent work has observed that Transformers may use several tokens as registers to aggregate global information (Darcet et al., 2023). To emphasize semantically relevant attention, we propose to normalize the answer-to-image attention of an image-question pair (x,q) by its value on a generic instruction q'. Specifically, we consider a fixed instruction q' ="Write a general description of the image.", and compute **relative attention** as $A_{rel}(x,q) = \frac{A_{si}(x,q)}{A_{si}(x,q')}$ under element-wise division. Fig. 2 shows examples of relative attention for LLaVA-1.5 and InstructBLIP (A_{rel}^{mk} at layers $m=14, k=0$ and $m=15, k=2$, respectively).

Do MLLMs Know Where to Look? Equipped with relative attention, we now return to our question of whether MLLMs have a localization limitation or perception limitation. To that end, we consider the validation set of TextVQA again. For each image-question pair, we first compute the relative attention. We then define **attention ratio** as the ratio of the total (sum) relative attention inside the answer ground-truth bounding box to its average across all bounding boxes of the same size

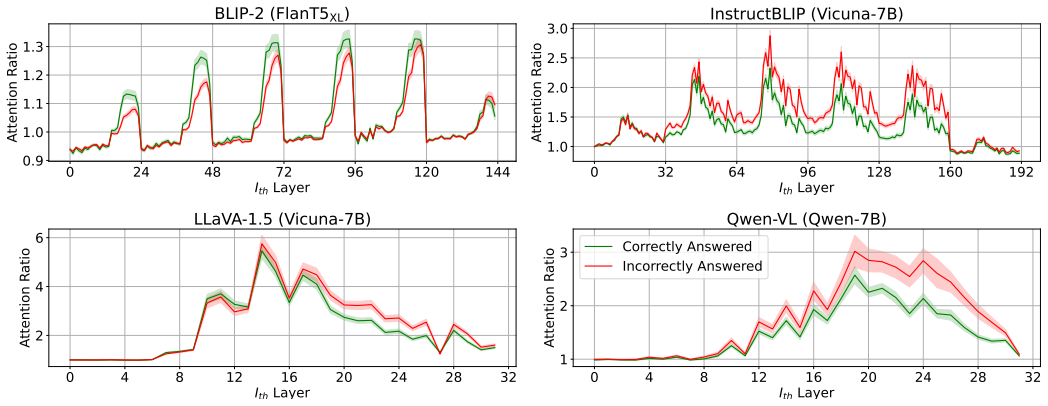

Figure 3: MLLMs' attention ratio across all layers (average with 95% CI over TextVQA). The attention ratio measures how significantly the MLLM is attending to the ground-truth bounding box (defined in Sec. 4). We observe that it is greater than 1 in most layers, showing that the MLLMs know where to look in the image even when they fail to answer correctly.

as the ground-truth on the image. This ratio provides a measure of how significantly the MLLM is attending to the ground-truth bounding box (in the sense of Markov's inequality). In Fig. 3, we plot the average (with 95% confidence interval) of the attention ratio, over the validation set of TextVQA for all layers in the considered MLLMs. The horizontal axis shows the combined layer index $l = m + k \times L$ for $m \in \{0 \dots L-1\}$ spanning the number of cross-attention layers in the backbone LLM, and $k \in \{0 \dots L_c - 1\}$ spanning the number of cross-attention layers in the connector (BLIP-2: $L = 24, L_c = 6$; InstructBLIP: $L = 32, L_c = 6$; Qwen-VL: $L = 32, L_c = 1$; LLaVA-1.5: $L = 32, L_c = 1$). In all MLLMs, we observe a significantly larger than 1 attention ratio in most layers, suggesting that the models are attending significantly to the ground-truth bounding box region on the image. Intriguingly, the models show similarly strong attention to the correct region regardless of whether they can answer the question correctly or incorrectly. These observations show that the MLLMs tend to know where to look, even when they answer incorrectly.

5 Automatic Visual Cropping (ViCrop)

We observed in Sec. 4 that the sensitivity of MLLMs to visual concept size is primarily a perception limitation (rather than a localization limitation). Therefore, one solution to mitigate this limitation is to simply train MLLMs with a larger number of image patches while maintaining per-patch resolution (hence increasing the image resolution of MLLMs). However, increasing the input image resolution by a factor of α, increases the number of ViT input patches (and output tokens) from N^2 to $\alpha^2 N^2$, which in turn increases the softmax attention computation complexity on the order of $\alpha^4 N^4$. Given that this solution is not scalable for current Transformer-based MLLMs, we choose to explore an alternative solution that **does not require any training and is scalable to any image resolution**. We note that several concurrent works have explored the first direction of *training* MLLMs with higher resolution image patches (Li et al., 2024c; Sun et al., 2024; Li et al., 2024d; McKinzie et al., 2024; Xu et al., 2024a; Luo et al., 2024), and notably LLaVA-Next (Liu et al., 2024a) has achieved the VQA state-of-the-art in several VQA benchmarks at the time of writing. We believe our work is parallel to these efforts in the following sense: rather than training higher and higher resolution MLLMs to enable them to see all resolutions (which is inevitably upper bounded), we explore how to smartly adjust the input image towards what an MLLM already can see without any additional training. We provide evidence showing that our training-free method can provide orthogonal benefits to the training-based methods in Appendices D and E.

Inspired by our findings that MLLMs tend to know where to look (Sec. 4) and that visual cropping can mitigate the perception limitation (Sec. 3), in this section we construct three automatic visual cropping methods in order to mitigate the perception limitation of MLLMs. These methods seek to use the internal information of an MLLM itself—in the form of attention maps and gradients—to find the approximate region of interest in images (*i.e.*, the region containing the subject of a question), and then to zoom into that region via visual cropping. One potential drawback of visual cropping is that some questions might need to have a global view of the image. To address this issue, we utilize the fact that MLLMs typically convert the image into a series of tokens. This allows us to directly

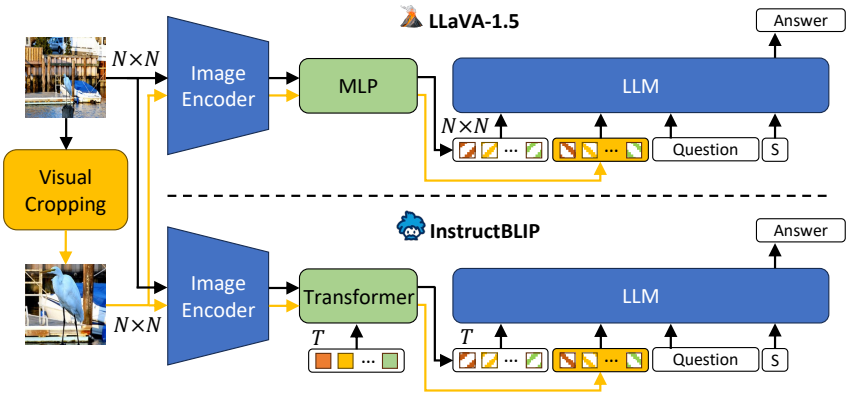

Figure 4: Illustration of the proposed visual cropping approach applied to two MLLMs.

extend the original image tokens by concatenating the visually cropped image tokens, as illustrated in Fig. 4. We use this concatenation approach when applying all our methods to MLLMs.

Relative Attention ViCrop (`rel-att`). In this method, we directly compute the relative attention $A_{rel}(x, q)$ defined in Sec. 4 for each image-question pair (x, q). We then select a target layer (in LLM and connector) based on a small held-out set of samples in TextVQA and use its relative attention as the importance map for visual cropping (discussed below). We ablate on the choice of layer in Sec. 6.

Gradient-Weighted Attention ViCrop (`grad-att`). The relative attention runs an additional generic instruction through the MLLM to normalize the answer-to-image attention and emphasize semantically relevant attention. As an alternative that does not require a second forward pass, we consider using the gradients to normalize attention, because the gradient of the model's decision with respect to an attention score shows how sensitive the decision is to changes in that attention, hence how semantically relevant the attention is for answering the question. To get a differentiable representation of the model's decision, we consider the logarithm of the maximum output probability at the starting answer token, $v = \log \text{softmax}(z(x, q))_{t^*} \in \mathbb{R}$, where $z \in \mathbb{R}^D$ is the output logit of the LLM at the starting answer position, D the vocabulary size, and $t^* = \arg\max_t z_t$. Then, following our notation in Sec. 4, we can compute the gradient-weighted versions of answer-to-token attention $\tilde{A}_{st}(x, q) = A_{st}(x, q) \odot \sigma(\nabla_{A_{st}} v(x, q))$ and token-to-image attention $\tilde{A}_{ti}(x, q) = A_{ti}(x) \odot \sigma(\nabla_{A_{ti}} v(x, q))$, where $\odot$ is element-wise product and $\sigma(w) = \max(0, w)$. We remove negative gradients because they correspond to tokens that if attended to will reduce the model certainty, hence often distracting locations Selvaraju et al. (2017). Finally, we compute the gradient-weighted answer-to-image attention as the following tensor product $\tilde{A}_{si}(x, q) = \tilde{A}_{st}(x, q) \otimes \tilde{A}_{ti}(x, q) \in \mathbb{R}^{L \times L_c \times 1 \times N^2}$. We will select the same target layer identified in `rel-att` from $\tilde{A}_{si}(x, q)$ as the importance map for visual cropping.

Input Gradient ViCrop (`pure-grad`). In this method, we seek to find the relevant regions on the image directly using the gradient of the MLLM's decision with respect to the input image. Compared to the previous attention-based methods, `pure-grad` is more versatile because it does not rely on the Transformer-based architecture. Specifically, for each image-question pair (x, q), we will compute $G(x, q) = \|\nabla_x v(x, q)\|_2$, where $v(x, q)$ is the logarithm of the maximum output probability of the LLM at the starting answer token (as defined in `grad-att` above), and the L2-norm is taken over the image channel dimension. However, gradients sometimes show high values in entirely constant-color regions (*e.g.*, blue skies). Given that these non-edge regions do not contain any visual details, we will explicitly diminish them in G. To that end, we first apply a 3×3-size Gaussian high-pass filter to the image, followed by a median filter of the same size to reduce salt-and-pepper noise. The resulting filtered image is then thresholded at its spatial median value to become a binary mask and is element-wise multiplied by G. Finally, the edge-emphasized G is spatially average-pooled into the $N \times N$ patches of the MLLM to become an importance map for visual cropping.

Bounding Box Selection for Visual Cropping. To convert the importance map (from each of the methods described above) to a bounding box, we use sliding windows of different sizes inspired by object detection literature Redmon et al. (2016). Specifically, for each MLLM, we define a set of windows with sizes equal to a multiple of the input image resolution of the MLLM. The multiples are in $\{1, 1.2, \dots 2\}$. We slide each window over the image with a stride of 1 and find its best position where the sum of the importance map inside the window is maximized. After selecting the best position per window, we choose the window whose internal sum has the largest difference from the average internal sum of its adjacent positions. This latter step is a heuristic to avoid choosing too small or too large windows (notice that in both cases, moving the window slightly left/right or up/down will not change its internal sum significantly). The chosen window is then cropped from the image, resized to the input image resolution of the MLLM, and provided to the MLLM in addition to the image-question pair.

High-Resolution Visual Cropping. In one of the benchmarks we consider in this work, V* Wu and Xie (2023), the images are of very high resolution (always more than 1K) and consequently, the resized input image provided to the MLLM might completely lose the visual concept of interest for a question. To mitigate this, on this benchmark, we employ a two-stage strategy. In the first stage, we divide images into non-overlapping blocks of smaller than 1024×1024 with an aspect ratio close to 1, compute the importance map separately for each block according to the ViCrop methods, and then re-attach the blocks back together. In the second stage, we find the bounding box for visual

Figure 5: Examples of `rel-att` helping MLLMs correct their mistakes (cyan-colored bounding box shows cropped region by `rel-att`; zoom-in insets are displayed for better readability).

cropping on this re-attached importance map exactly as described before and provide the original image-question pair with the resized cropped image to the MLLM.

6 ViCrop Method Analysis

In this section, we apply our proposed visual cropping methods to two open-source SOTA MLLMs, InstructBLIP (Vicuna-7B) (Dai et al., 2023) and LLaVA-1.5 (Vicuna-7B) (Liu et al., 2023a). We evaluate their effectiveness in improving the perception of smaller visual concepts on 4 detail-sensitive datasets (TextVQA [2] (Singh et al., 2019), V* (Wu and Xie, 2023), POPE (Li et al., 2023c), DocVQA (Mathew et al., 2021)), and their ability to maintain performance on larger visual concepts in 3 general-purpose datasets containing mostly large objects (GQA (Hudson and Manning, 2019), AOKVQA (Schwenk et al., 2022), VQAv2 (Goyal et al., 2017)). InstructBLIP uses the hyper-parameters $N = 16, m = 15, k = 2$ and input image resolution of 224×224. LLaVA-1.5 uses $N = 24, m = 14$ and input image resolution of 336×336. When reporting accuracy, we compute VQA-score[3] for all benchmarks except GQA. For GQA, we compute accuracy using the official code.[4]. See Appendices A to C for further details about implementation, datasets, and prompts.

ViCrop Improves VQA Accuracy. In Fig. 5, we show a few examples of the ViCrop helping the MLLM correct itself (more examples in Appendix G), and in Tab. 2, we report the accuracy of the proposed ViCrop methods on the VQA benchmarks. We observe that all methods significantly improve the accuracy of the original MLLMs (*no cropping*) on detail-sensitive benchmarks, without requiring any training, while maintaining the MLLMs' performance on benchmarks with larger visual concepts. Thus, the accuracy gain on fine details (most notably in TextVQA and V*) does not seem to come at the cost of accuracy on larger visual details and relations. We also observe that the accuracy gain for LLaVA-1.5 is more substantial than for InstructBLIP. This can be explained by the

[2]†In TextVQA evaluation, we do not provide externally extracted OCR tokens to the MLLM since we want to measure its true perception, this differs from the setup in the original paper. See more discussion in Appendix A.

[3]https://visualqa.org/evaluation.html

[4]https://cs.stanford.edu/people/dorarad/gqa/evaluate.html

Table 2: Accuracy of the proposed ViCrop methods on visual question answering benchmarks.

Model		Smaller Visual Concepts				Larger Visual Concepts		
		TextVQA†	V*	POPE	DocVQA	AOKVQA	GQA	VQAv2
LLaVA-1.5	no cropping	47.80	42.41	85.27	15.97	59.01	60.48	75.57
	`rel-att`	55.17	**62.30**	**87.25**	19.63	**60.66**	60.97	**76.51**
	`grad-att`	**56.06**	57.07	87.03	**19.84**	59.94	**60.98**	76.06
	`pure-grad`	51.67	46.07	86.06	17.70	59.92	60.54	75.94
InstructBLIP	no cropping	33.48	35.60	84.89	9.20	60.06	49.41	76.25
	`rel-att`	45.44	**42.41**	86.64	9.95	61.28	49.75	**76.84**
	`grad-att`	**45.71**	37.70	**86.99**	**10.81**	**61.77**	**50.33**	76.08
	`pure-grad`	42.23	37.17	86.84	8.99	61.60	50.08	76.71

Table 3: Ablation study on the choice of layer and the use of high-resolution visual cropping.

Model		Choice of Layer			High-Resolution ViCrop		
		Selective	Average	Δ	w/ High-Res	w/o High-Res	Δ
LLaVA-1.5	no cropping	47.80	–	–	42.41	42.41	–
	`rel-att`	55.17	55.45	+0.28	62.30	47.64	-14.66
	`grad-att`	56.06	56.26	+0.20	57.07	49.74	-7.33
	`pure-grad`	51.67	–	–	46.07	45.03	-1.04
InstructBLIP	no cropping	33.48	–	–	35.60	35.60	–
	`rel-att`	45.44	44.40	-1.04	42.41	38.74	-3.67
	`grad-att`	45.71	44.98	-0.73	37.70	42.41	+4.71
	`pure-grad`	42.23	–	–	37.17	42.41	+5.24

fact that InstructBLIP only trains its connector and not its backbone LLM during tuning—the LLM does not adapt to use the image tokens, rather the image tokens are adapted to optimally prompt the LLM—and therefore the LLM cannot effectively use the additional (cropped) image tokens provided through visual cropping. Nonetheless, the results show that ViCrop can be effectively applied to different MLLMs, and is a promising inference-time solution for mitigating the perception limitation observed in Sec. 3.

Ablation Study on the Choice of Layer. To understand the importance of the choice of an informative layer for `rel-att` and `grad-att` (as discussed in Sec. 5), in Tab. 3 we compare the accuracy of these methods when simply taking the average of all layers in A_{rel} and $\tilde{A}_{si}$, respectively, on TextVQA. We observe that `rel-att` is robust to this choice and `grad-att` declines about 3.5 percentage points in accuracy. Importantly, both methods still improve the MLLMs' accuracy even when using the layer average, suggesting that averaging is a suitable choice in the absence of any data for selecting a layer.

Ablation Study on the High-Resolution ViCrop. In Sec. 5, we proposed a two-stage strategy for processing the very high-resolution images in the V* benchmark. To see how effective this strategy is, in Tab. 3 we compare the accuracy of ViCrop methods with and without this high-resolution strategy on V*. We observe that while this strategy is very beneficial to LLaVA-1.5, it declines the performance of `grad-att` and `pure-grad` for InstructBLIP. However, all methods, with and without this strategy, still improve the MLLMs' accuracy.

ViCrop with External Tools. In addition to the internal ViCrop methods, we also considered the use of external off-the-shelf models to find the region of interest in an image for visual cropping. Specifically, we utilized SAM (Kirillov et al., 2023), YOLO (Redmon et al., 2016), and CLIP (Radford et al., 2021) to find the most relevant part of an image to a given question (details of these external ViCrop methods are provided in Appendix F). In Tab. 4, we compare the accuracy of external ViCrop methods to the internal methods on TextVQA. While external models are also effective in improving the accuracy of MLLMs, they are weaker than all the proposed internal ViCrop methods, thus we did not explore them further.

Table 4: Accuracy of ViCrop using external tools compared to attention/gradient (on TextVQA); and the inference time overhead of ViCrop methods (in seconds). Original's time is per answer token.

	Model	Original	SAM	YOLO	CLIP	`rel-att`	`grad-att`	`pure-grad`
Accuracy (TextVQA)	LLaVA-1.5	47.80	49.42	48.84	48.55	55.17	56.06	51.67
	InstructBLIP	33.48	39.23	36.49	39.61	45.44	45.71	42.23
CPU Time	LLaVA-1.5	2.26	91.53	0.97	5.46	14.43	11.33	29.86
	InstructBLIP	0.66				4.35	3.78	7.04
GPU Time	LLaVA-1.5	0.17	3.33	0.35	1.07	1.16	0.89	2.36
	InstructBLIP	0.06				0.28	0.29	0.60

Inference Time Overhead. In Tab. 4, we report the average inference-time overhead of the proposed visual cropping methods on GPU (NVIDIA RTX A6000) and CPU (Intel(R) Gold 5317 CPU @ 3.00GHz) and compare with the per-answer-token processing time of the MLLMs. We see that all proposed methods (except SAM) are reasonably fast (1 to 2 seconds overhead on GPU). For example, computing the visual cropping with `rel-att` takes the time of generating only 5 tokens by the MLLM. **Note that our methods' time overhead will not scale with the number of answer tokens and is constant regardless of how long the answer is** because our external methods do not need any answer token, and internal methods only need the starting answer token (see Sec. 5). In contrast, MLLMs' inference time scales approximately linearly with the number of answer tokens.

7 Conclusion

In this work, we qualitatively and quantitatively showed that there exists a perception bias against small visual details in widely-used MLLMs. Then we found that MLLMs often know where to look even if they fail to answer the question, indicating that the bias toward small visual details is rooted in a perception limitation rather than a localization limitation. To mitigate this limitation, we proposed multiple automatic visual localization methods as scalable and training-free solutions based on models' internal dynamics while answering the visual questions. Through evaluation of multiple multimodal benchmarks, we showed that our method can significantly improve MLLMs' accuracy without requiring any training, especially in detail-sensitive scenarios. Our findings suggest that MLLMs should be used with caution in detail-sensitive applications, and that visual cropping/localization with the model's own knowledge is a promising direction to enhance their performance.

Limitations and Future Work. The proposed ViCrop methods do not enhance all types of questions equally. We have observed that questions concerning relations and counting are particularly difficult for ViCrop methods to help answer. This is expected as the proposed ViCrop can only focus on one region in the image. We leave extending ViCrop to focus on multiple regions simultaneously for future work. Another limitation of the proposed methods is their time overhead and the addition of visual tokens. While the overhead is reasonable (a few seconds), we believe it can be significantly optimized as an inference-time mechanism, for example by utilizing lower precision, and weight quantization. Furthermore, Matryoshka Query Transformer (MQT) (Hu et al., 2024) enables MLLMs to have varying visual context size during inference. In our current results, we have shown that our methods can work with two different MLLMs with distinct visual context sizes, so it seems entirely possible that our method can still work with varying visual context size under MQT, which can further reduce our computational cost through rescaling the cropped image. We leave these inference cost optimizations to future works. Lastly, we have observed that the proposed methods tend to have some complementary benefits, and therefore exploring ways to combine them (for example based on the prediction uncertainty) is also a very interesting direction for future research.

ACKNOWLEDGMENTS

We thank Jinyi Hu and Joe Mathai for their very useful insights. We also express our gratitude to anonymous reviewers for their valuable feedback. This research was supported in part by the National Science Foundation under Contract No. IIS-2153546.

REFERENCES

Anthropic. The claude 3 model family: Opus, Sonnet, Haiku, March 2024. URL https://www-cdn.anthropic.com/de8ba9b01c9ab7cbabf5c33b80b7bbc618857627/Model_Card_Claude_3.pdf.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 2023.

Boyuan Chen, Zhuo Xu, Sean Kirmani, Brain Ichter, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14455–14465, June 2024.

Liang Chen, Lei Li, Haozhe Zhao, Yifan Song, and Vinci. R1-v: Reinforcing super generalization ability in vision-language models with less than $3. https://github.com/Deep-Agent/R1-V, 2025. Accessed: 2025-02-02.

Prateek Chhikara, Dhiraj Chaurasia, Yifan Jiang, Omkar Masur, and Filip Ilievski. Fire: Food image to recipe generation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 8184–8194, 2024.

Jaemin Cho, Jie Lei, Hao Tan, and Mohit Bansal. Unifying vision-and-language tasks via text generation. In *International Conference on Machine Learning*, pages 1931–1942. PMLR, 2021.

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.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. *arXiv preprint arXiv:2309.16588*, 2023.

Mostafa Dehghani, Basil Mustafa, Josip Djolonga, Jonathan Heek, Matthias Minderer, Mathilde Caron, Andreas Steiner, Joan Puigcerver, Robert Geirhos, Ibrahim M Alabdulmohsin, et al. Patch n'pack: Navit, a vision transformer for any aspect ratio and resolution. *Advances in Neural Information Processing Systems*, 36:2252–2274, 2023.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

Jiahui Gao, Renjie Pi, Jipeng Zhang, Jiacheng Ye, Wanjun Zhong, Yufei Wang, Lanqing Hong, Jianhua Han, Hang Xu, Zhenguo Li, et al. G-llava: Solving geometric problem with multi-modal large language model. *arXiv preprint arXiv:2312.11370*, 2023.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6904–6913, 2017.

Jiaxian Guo, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Boyang Li, Dacheng Tao, and Steven Hoi. From images to textual prompts: Zero-shot visual question answering with frozen large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10867–10877, 2023.

Tanmay Gupta and Aniruddha Kembhavi. Visual programming: Compositional visual reasoning without training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14953–14962, 2023.

Wenbo Hu, Zi-Yi Dou, Liunian Harold Li, Amita Kamath, Nanyun Peng, and Kai-Wei Chang. Matryoshka query transformer for large vision-language models. *arXiv preprint arXiv:2405.19315*, 2024.

Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709, 2019.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Glenn Jocher, Ayush Chaurasia, and Jing Qiu. YOLO by Ultralytics. January 2023. URL https://github.com/ultralytics/ultralytics.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, et al. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024a.

Chunyuan Li, Cliff Wong, Sheng Zhang, Naoto Usuyama, Haotian Liu, Jianwei Yang, Tristan Naumann, Hoifung Poon, and Jianfeng Gao. Llava-med: Training a large language-and-vision assistant for biomedicine in one day. *arXiv preprint arXiv:2306.00890*, 2023a.

Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34:9694–9705, 2021.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pages 12888–12900. PMLR, 2022a.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023b.

Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10965–10975, 2022b.

Xiang Li, Cristina Mata, Jongwoo Park, Kumara Kahatapitiya, Yoo Sung Jang, Jinghuan Shang, Kanchana Ranasinghe, Ryan Burgert, Mu Cai, Yong Jae Lee, et al. Llara: Supercharging robot learning data for vision-language policy. *arXiv preprint arXiv:2406.20095*, 2024b.

Yanwei Li, Yuechen Zhang, Chengyao Wang, Zhisheng Zhong, Yixin Chen, Ruihang Chu, Shaoteng Liu, and Jiaya Jia. Mini-gemini: Mining the potential of multi-modality vision language models. *arXiv preprint arXiv:2403.18814*, 2024c.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*, 2023c.

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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26763–26773, 2024d.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023a.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023b.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, January 2024a. URL https://llava-vl.github.io/blog/2024-01-30-llava-next/.

Shilong Liu, Hao Cheng, Haotian Liu, Hao Zhang, Feng Li, Tianhe Ren, Xueyan Zou, Jianwei Yang, Hang Su, Jun Zhu, et al. Llava-plus: Learning to use tools for creating multimodal agents. In *European Conference on Computer Vision*, pages 126–142. Springer, 2024b.

Gen Luo, Yiyi Zhou, Yuxin Zhang, Xiawu Zheng, Xiaoshuai Sun, and Rongrong Ji. Feast your eyes: Mixture-of-resolution adaptation for multimodal large language models. *arXiv preprint arXiv:2403.03003*, 2024.

Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2200–2209, 2021.

Brandon McKinzie, Zhe Gan, Jean-Philippe Fauconnier, Sam Dodge, Bowen Zhang, Philipp Dufter, Dhruti Shah, Xianzhi Du, Futang Peng, Floris Weers, et al. Mm1: Methods, analysis & insights from multimodal llm pre-training. *arXiv preprint arXiv:2403.09611*, 2024.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. A-okvqa: A benchmark for visual question answering using world knowledge. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part VIII*, pages 146–162. Springer, 2022.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

Hao Shao, Shengju Qian, Han Xiao, Guanglu Song, Zhuofan Zong, Letian Wang, Yu Liu, and Hongsheng Li. Visual cot: Unleashing chain-of-thought reasoning in multi-modal language models, 2024.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8317–8326, 2019.

Hai-Long Sun, Da-Wei Zhou, Yang Li, Shiyin Lu, Chao Yi, Qing-Guo Chen, Zhao Xu, Weihua Luo, Kaifu Zhang, De-Chuan Zhan, et al. Parrot: Multilingual visual instruction tuning. *arXiv preprint arXiv:2406.02539*, 2024.

Dídac Surís, Sachit Menon, and Carl Vondrick. Vipergpt: Visual inference via python execution for reasoning. *arXiv preprint arXiv:2303.08128*, 2023.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024.

Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, et al. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*, 2025.

Anthony Meng Huat Tiong, Junnan Li, Boyang Li, Silvio Savarese, and Steven CH Hoi. Plug-and-play vqa: Zero-shot vqa by conjoining large pretrained models with zero training. *arXiv preprint arXiv:2210.08773*, 2022.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, et al. Image as a foreign language: Beit pretraining for all vision and vision-language tasks. *arXiv preprint arXiv:2208.10442*, 2022.

Penghao Wu and Saining Xie. V*: Guided visual search as a core mechanism in multimodal llms. *arXiv preprint arXiv:2312.14135*, 2023.

Ruyi Xu, Yuan Yao, Zonghao Guo, Junbo Cui, Zanlin Ni, Chunjiang Ge, Tat-Seng Chua, Zhiyuan Liu, Maosong Sun, and Gao Huang. Llava-uhd: an lmm perceiving any aspect ratio and high-resolution images. *arXiv preprint arXiv:2403.11703*, 2024a.

Zhenhua Xu, Yujia Zhang, Enze Xie, Zhen Zhao, Yong Guo, Kwan-Yee K Wong, Zhenguo Li, and Hengshuang Zhao. Drivegpt4: Interpretable end-to-end autonomous driving via large language model. *IEEE Robotics and Automation Letters*, 2024b.

Tianyu Yu, Yuan Yao, Haoye Zhang, Taiwen He, Yifeng Han, Ganqu Cui, Jinyi Hu, Zhiyuan Liu, Hai-Tao Zheng, Maosong Sun, et al. Rlhf-v: Towards trustworthy mllms via behavior alignment from fine-grained correctional human feedback. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13807–13816, 2024.

Jiarui Zhang, Filip Ilievski, Kaixin Ma, Aravinda Kollaa, Jonathan Francis, and Alessandro Oltramari. A study of situational reasoning for traffic understanding. *KDD*, 2023a.

Jiarui Zhang, Mahyar Khayatkhoei, Prateek Chhikara, and Filip Ilievski. Visual cropping improves zero-shot question answering of multimodal large language models. In *R0-FoMo: Robustness of Few-shot and Zero-shot Learning in Large Foundation Models*, 2023b.

Jiarui Zhang, Jinyi Hu, Mahyar Khayatkhoei, Filip Ilievski, and Maosong Sun. Exploring perceptual limitation of multimodal large language models. *arXiv preprint arXiv:2402.07384*, 2024a.

Jiarui Zhang, Ollie Liu, Tianyu Yu, Jinyi Hu, and Willie Neiswanger. Euclid: Supercharging multimodal llms with synthetic high-fidelity visual descriptions. *arXiv preprint arXiv:2412.08737*, 2024b.

Renrui Zhang, Xinyu Wei, Dongzhi Jiang, Ziyu Guo, Shicheng Li, Yichi Zhang, Chengzhuo Tong, Jiaming Liu, Aojun Zhou, Bin Wei, et al. Mavis: Mathematical visual instruction tuning with an automatic data engine. *arXiv preprint arXiv:2407.08739*, 2024c.

Tiancheng Zhao, Tianqi Zhang, Mingwei Zhu, Haozhan Shen, Kyusong Lee, Xiaopeng Lu, and Jianwei Yin. Vl-checklist: Evaluating pre-trained vision-language models with objects, attributes and relations. *arXiv preprint arXiv:2207.00221*, 2022.

A Implementation Details

We use *python 3.10.6, transformers 4.29.1 and torch 2.1.2* for all the experiments. Our environment consists of an Intel(R) Gold 5317 CPU @ 3.00GHz with 48 cores and 756 GB of RAM, and we utilize NVIDIA RTX A6000 GPUs for our experiments. We use the huggingface implementations of all studied MLLMs with the recommended hyper-parameters according to the respective papers. For GPT-4o, we use the official public API, which is available at the time of submission.

Regarding the evaluation setting of the TextVQA dataset in Tab. 2, our setting is slightly different from the one used by the LLaVA-1.5 original paper Liu et al. (2023a). They report accuracy on TextVQA by using externally extracted OCR tokens to enrich its text prompt. This is a text-specific trick that essentially out-sources the perception of text to an external OCR model. This text-specific trick is not mentioned in their paper or supplementary material, but see their clarification in response to a GitHub issue here: `https://github.com/haotian-liu/LLaVA/issues/515#issuecomment-1763779341`. In contrast, we treat TextVQA the same as any other vision dataset in our experiments, that is, we do not provide any OCR extracted tokens to MLLMs when applying them to TextVQA (only image and question, in the evaluation prompt format specified in their respective papers). This results in a slightly lower accuracy compared to the one reported in the original paper, but instead, this number shows the true perception ability of LLaVA-1.5 on TextVQA, not confounded by the ability of an external OCR model. For completeness, we also measured TextVQA accuracy in the presence of OCR tokens, which results in 59.8 for LLaVA-1.5 without any visual cropping, and 63.95 with `rel-att`, showing that our proposed visual cropping can still be beneficial even when OCR tokens are provided to the MLLM.

B Dataset Statistics

In this section, we present the details of the datasets used for evaluation in this paper. We report the average height and weight of the images in the dataset. We also report the number of images and questions in each dataset. For VQAv2, we run our experiment on a random 50K subset of the official validation set. We use the entire validation set in all other datasets.

Table 5: Average width ($\bar{W}$) and height ($\bar{H}$) of images, number of images, and number of questions on all datasets.

	V*	DocVQA	TextVQA	POPE	AOKVQA	GQA	VQAv2
$\bar{W}$	2246	1776	954	584	581	578	577
$\bar{H}$	1582	2084	818	478	480	482	485
# Images	191	1286	3166	500	1122	398	14206
# Questions	191	5349	5000	8910	1145	10781	50000

For our analysis presented in Table 1 and Figure 3, we focused on TextVQA dataset, which includes bounding box annotations for OCR-detected text within images. However, this dataset does not specify which bounding boxes correspond to the regions where answers are located, necessitating a manual annotation process. The TextVQA dataset comprises 5000 questions and 3166 images. We manually annotated these question-image pairs, ensuring accurate bounding boxes over **all the regions** of interest where the answers could be found. This manual annotation process was essential for our analysis, allowing us to provide precise and reliable ground-truth data for the study. Given that some questions were associated with multiple bounding boxes in their corresponding images, we undertook a filtering process to isolate the question-image pairs. This effort resulted in a refined set of 4370 question-image pairs, where there is only one instance of the subject of the question in the image. For example, if the question is "what type of drink is sold here?" and there are two different cans of drinks in the image, we remove this image-question pair.

C PROMPT FORMAT FOR ZERO-SHOT INFERENCE

In this section, we provide details about the prompt format used in models for zero-shot inference. We use a different prompt format for LLaVA and InstructBLIP which we adapt from the original papers, as shown below.

LLaVA-1.5

```
<image> USER:{question} Answer the question using a single
word or phrase.  ASSISTANT:
```

InstructBLIP

```
<image> Question:{question} Short Answer:
```

D ORTHOGONAL BENEFITS TO LLAVA-NEXT

We apply our proposed `rel-att` visual cropping method to an additional newer MLLM – LLaVA-NeXT (Liu et al., 2024a) current SOTA in several VQA benchmarks – that has support for higher-resolution compared to LLaVA-1.5. In Tab. 6, we observe that our method can still boost the MLLM's performance, without requiring any training. This provides further evidence for the generalizability of our proposed visual cropping and its orthogonal benefits to training MLLMs with higher image patch resolution.

Table 6: Orthogonal benefits of visual cropping when applied to LLaV-NeXT that is trained to adapt to processing high-resolution images.

Model	TextVQA	V*
LLaVA-NeXT (Mistral-7B)	65.17	58.11
LLaVA-NeXT (Mistral-7B) + `rel-att`	68.65	61.78

E COMPARISON WITH THE V* METHOD (SEAL)

The V* method (SEAL) (Wu and Xie, 2023) proposes a multi-agent fine-tuning approach to enhance the ability of an underlying MLLM to answer questions about small visual concepts. However, SEAL requires substantial training and finetuning of several neural networks, whereas our methods are completely training-free, so a direct comparison would not be fair. Nonetheless, to provide an idea of how our method compares to SEAL in an "as-is" fashion (i.e. if a user just wants to pick one method as-is off-the-shelf), we report the accuracy of SEAL compared to LLaVA-1.5+`rel-att` in Tab. 7. We observe that our method outperforms SEAL except on the V* benchmark. We think this might be because SEAL is designed and tuned specifically toward high-resolution images in its V* benchmark. We also note that the inference time of SEAL is slower than our method (4.44s compared to 1.88s on average per question, tested on the same random 100 TextVQA samples with one A6000 GPU). That being said, we note that our methods and SEAL can both help enhance MLLMs, and our methods can be integrated into SEAL or other multi-agent pipelines.

Table 7: Performance comparison between our `rel-att` applied on LLaVA-1.5 and SEAL (Wu and Xie, 2023) across multiple vision-language benchmarks.

Model	TextVQA	V*	POPE	DocVQA	AOKVQA	GQA	VQAV2
SEAL	36.30	75.30	82.40	5.31	55.34	50.18	65.35
LLaVA-1.5+`rel-att`	55.17	62.30	87.25	19.63	60.66	60.97	76.29

F External Tools ViCrop

In this section, we present three automatic question-guided localization methods based on popular off-the-shelf vision-based models, namely CLIP Radford et al. (2021), YOLO Redmon et al. (2016), and SAM Kirillov et al. (2023). These three methods utilize external vision-based knowledge for the localization process through multimodal encoding, object detection, and semantic segmentation, respectively. See Tab. 4 for their results compared to internal ViCrop methods.

CLIP ViCrop. The intuition of this method is to progressively refine the image towards the region of highest relevance to a given question using CLIP Radford et al. (2021). CLIP consists of an image encoder and a text encoder, which are trained on a large dataset of image-caption pairs to map each image (caption) close to its caption (image) and far from all other captions (images). The result is an aligned shared space where various images can be directly compared with various texts. To find the region of interest, given an image-question pair, we first crop the image from the four sides (top, bottom, left, and right) at a cropping ratio of 0.9 to produce four overlapping cropped images. We then use CLIP to assess the semantic similarity between these cropped images and the question. The highest-scoring crop is chosen as the input for the next iteration. This process is repeated for 20 iterations, and the cropped image with the highest CLIP similarity to the question is selected for visual cropping.

YOLO ViCrop. Instead of a progressive approach to finding the region of interest, in this method we select candidate regions based on a state-of-the-art object detection method: YOLOv8 (Jocher et al., 2023) pretrained on COCO Lin et al. (2014). Using YOLO, we filter out regions that contain no salient objects – *i.e.*, regions for which CLIP could mistakenly assign high similarity. More concretely, for each question-image pair, we first use YOLO to collect bounding boxes for all predicted objects with confidence higher than 0.25 (the recommended default).[5] Then, for each predicted bounding box, we crop its corresponding image and compute its similarity to the question using CLIP. Finally, the bounding box with the highest similarity score is selected as the region of interest for visual cropping.

SAM ViCrop. A limitation of YOLO is that it only provides bounding boxes corresponding to a fixed number of object classes. To overcome this issue, we use the segment anything model (SAM) Kirillov et al. (2023), which has shown state-of-the-art zero-shot segmentation performance. SAM can provide an extensive set of segmentation masks for each image, thus providing a more granular set of salient candidate regions compared to YOLO. More concretely, for each image-question pair, we feed the image into SAM, which provides an extensive set of segmentation masks corresponding to all objects and object parts. Then, we translate these masks into bounding boxes by computing the smallest bounding box that covers each segmentation mask. Finally, the bounding box with the highest CLIP similarity to the question is selected as the region of interest for visual cropping.

Finally, for each method, we crop the smallest covering square (so that the cropped image is not deformed when resized to the input resolution of the MLLM), and provide it to the MLLM in addition to the original image-question pair (as depicted in Fig. 4).

[5] https://docs.ultralytics.com/modes/predict

G ADDITIONAL EXAMPLES ON MODEL'S PREDICTIONS

Figure 6: Success (first 3) and failure (last) examples of LLaVA-1.5 (`rel-att`) on the V* benchmark (cyan-colored bounding box shows cropped region by `rel-att`; zoom-in insets are displayed for better readability).

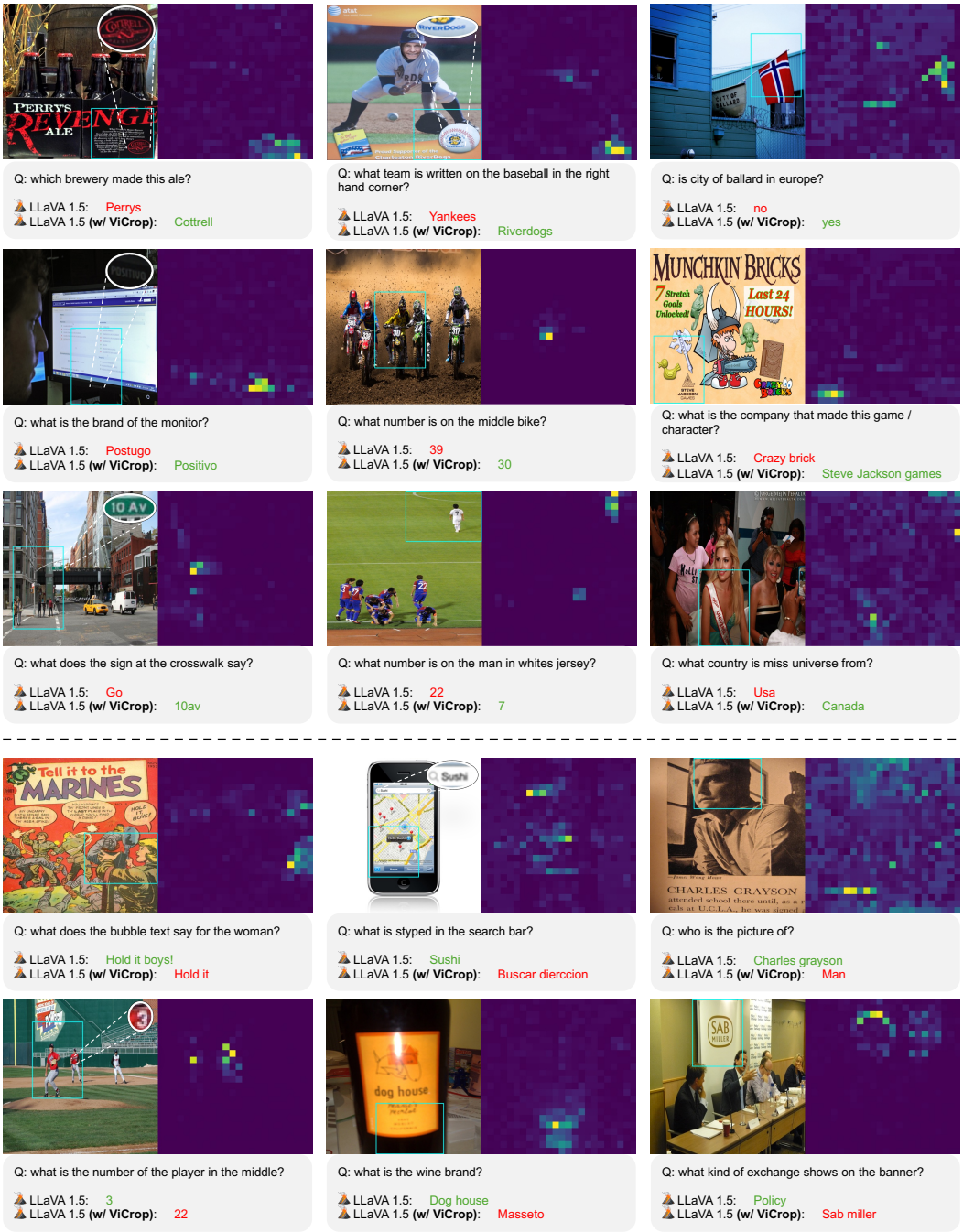

Figure 7: Success (first 9) and failure (last 6) examples of LLaVA-1.5 (`rel-att`) on the TextVQA benchmark (cyan-colored bounding box shows cropped region by `rel-att`).

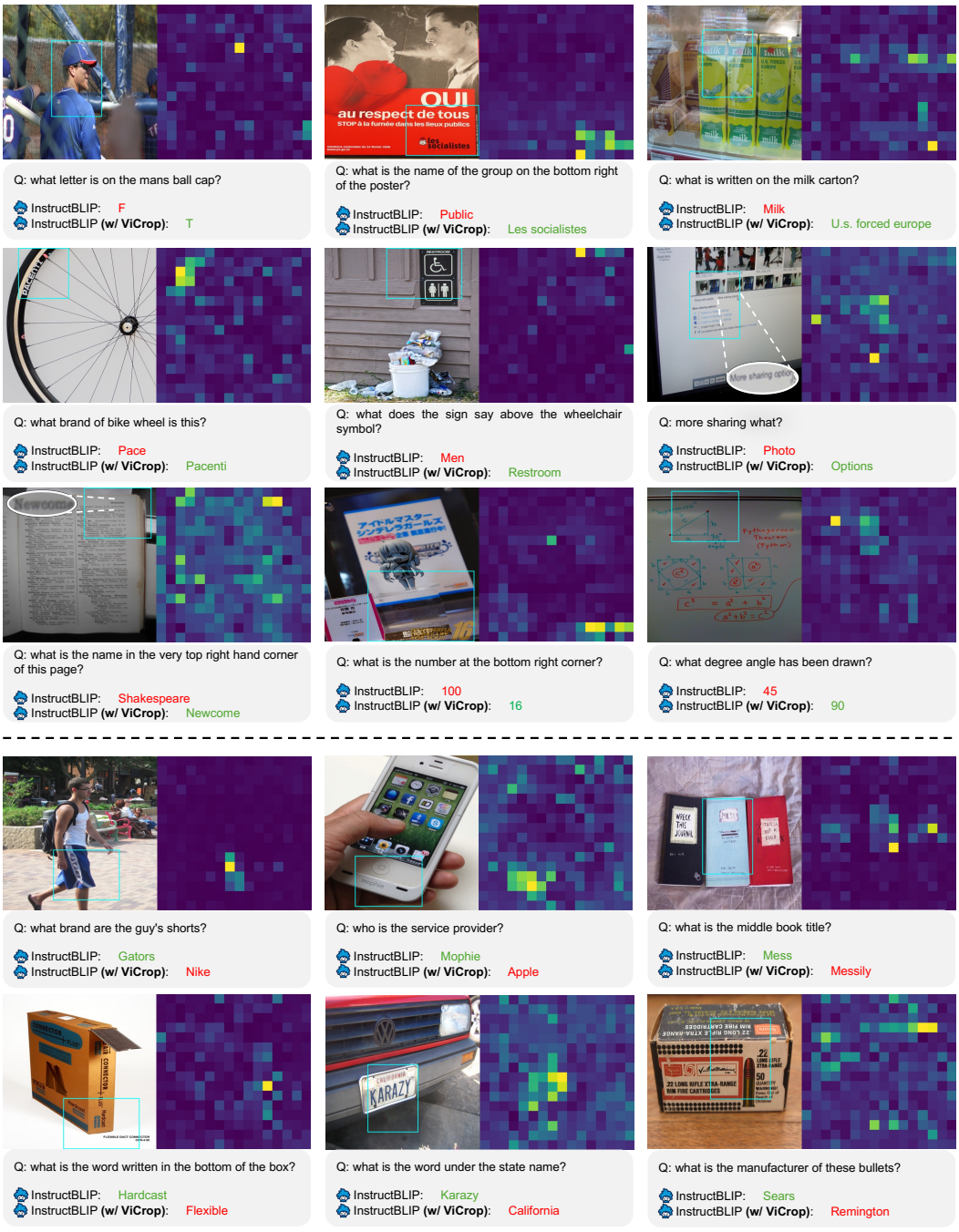

Figure 8: Success (first 9) and failure (last 6) examples of InstructBLIP (`rel-att`) on the TextVQA benchmark (cyan-colored bounding box shows cropped region by `rel-att`).