Draft and Refine with Visual Experts

Supplementary Material

7. Implementation Detail

7.1. Large Vision Language Models

All backbones are used in their instruction-tuned form, remain frozen, and run in half precision (float16/bfloat16) on a single A100 GPU. We use the following models:

- **Idefics-9B** (`idefics-9b-instruct`): vision–text encoder–decoder with a 224×224 vision input.
- **PaliGemma-3B** (`paligemma-3b-448px`): multimodal encoder–decoder with a 448×448 vision tower.
- **InstructBLIP-7B** (`instructblip-vicuna-7b`): Vicuna-7B decoder with a learnable vision projector using 224×224 inputs.
- **LLaVA-1.6 Mistral-7B** (`llava-1.6-mistral-7b`): Mistral-based LVLM with a CLIP ViT-L/14 backbone. The model internally uses the original 336×336 CLIP resolution and performs multiple crop-based views for enhanced grounding.
- **Qwen2.5-VL-7B** (`qwen2.5-vl-7b-instruct`): unified vision–language transformer with a high-resolution vision encoder (dynamic resolution).
- **CogVLM** (`cogvlm-chat`): Vicuna-based causal decoder with a multimodal vision tower using 336×336 inputs.
- **MiniGPT-v2** (`minigpt-v2`): lightweight generative LVLM coupling a pretrained vision encoder with a 448×448 input resolution.

All models operate in half precision; InstructBLIP and LLaVA require float16 for stable decoding, while the others use bfloat16.

7.2. Query Generation

We construct question-conditioned query sets using a large language model. For each image–question pair in the evaluation split, an LLM generates a compact reformulation of the question that captures its semantic intent. All datasets are loaded with their standard validation or test partitions, and the corresponding data loader returns image batches and raw questions. The query for each sample is produced by feeding the question into an instruction-tuned LLM together with optional dataset-specific prompts. We use the following configuration:

- Queries are generated with Llama 3 (70B), executed on a multi GPUs.
- For each batch, the LLM receives the raw question and outputs a refined textual query reflecting the core information need.
- Each output query is stored as a text unit paired with the original question and used as input to our relevance estimation module.

The resulting queries serve as compact, LLM-generated representations of the information sought by each question and are used to compute question-conditioned relevance maps in subsequent stages.

> Extract the most important visual keywords (objects, attributes, or actions)
> from the following question. Return them as a comma-separated list.
> Question: {q}
> Keywords:

7.3. Relevance Map

We estimate pixel-level relevance using CLIPSeg. For each image and its associated textual phrases, CLIPSeg produces a set of segmentation logits that indicate how strongly each phrase corresponds to each location in the image. The relevance estimation pipeline is as follows:

- We use the CLIPSeg `rd64-refined` model to compute phrase-conditioned segmentation scores.
- For VQA tasks, the input text consists of the LLM-generated query phrases; for captioning, we use the 80 COCO object categories as textual prompts.
- Each image is converted to RGB, processed together with its list of phrases, and passed through CLIPSeg to obtain a stack of logits of shape $N \times H \times W$, where N is the number of phrases.
- The logits are transformed with a sigmoid, and the per-phrase maps are aggregated by summation to form a single relevance map.
- The aggregated map is resized to 336×336 using bilinear interpolation.
- Finally, the map is normalized with a spatial softmax over all pixels, producing a probability distribution that highlights regions most relevant to the input phrases.

The resulting relevance map serves as the question- or object-conditioned spatial prior used in subsequent masking and perturbation steps.

7.4. Experts

We employ a set of frozen visual experts to extract complementary signals used in the Refine stage. All models run in half precision and operate independently of the backbone LVLM.

- **GroundingDINO** (`mm_grounding_dino_large_all`): a zero-shot object detector that predicts bounding boxes conditioned on textual phrases.

- **SAM** (`sam-vit-base`): a segmentation model that produces dense object masks given an input image.
- **Depth Anything V2** (`depth-anything-v2-small`): a monocular depth estimator that outputs a dense depth map for each image.
- **MDETR / Deformable-DETR** (`deformable-detr`): a transformer-based grounding model that aligns textual descriptions with object regions.

Each expert provides complementary structural cues including object boxes, masks, depth information, and grounded regions, which are used by the Refine stage to improve evidence extraction.

7.5. Rendering

All experts support a unified set of visualization modes during rendering: **black** (retain selected regions on a black background), **blur** (blur non-selected regions), **gray** (desaturate background), **white** (replace background with white), and **highlight** (increase brightness and contrast of selected regions). These modes provide consistent interpretability across detection, segmentation, depth estimation, and referring expression grounding.

7.5.1. Object Detection: GroundingDINO

We implement GroundingDINO [40] as the detection expert for prompt-based region extraction. The module processes a batch of images together with natural language prompts. In the absence of a prompt, the system defaults to a detect-all mode. Images are encoded with the official processor, and model outputs are post-processed into bounding boxes, labels, and confidence scores. A threshold-based refinement step filters low-confidence predictions and produces both structured JSON summaries and textual descriptions. Detected regions are then visualized using the unified rendering modes.

7.5.2. Segmentation: Segment Anything

We incorporate the Segment Anything Model (SAM) [22] for segmentation-based expert reasoning. SAM requires no textual input and produces multiple candidate masks with associated scores. The masks are aggregated into a single binary segmentation mask, which is rendered on top of the original image using the same visualization modes as the detection expert.

7.5.3. Depth Estimation: Depth Anything v2

Depth Anything v2 [69] serves as the depth-estimation expert. Given an input image, the model predicts a dense depth map, which is normalized and resized to match the original spatial resolution. For visualization, the depth estimate is converted into a grayscale map and blended with the input RGB image. The blending ratio is user-controlled, ranging from pure RGB to pure depth representation.

Dataset	Total Samples	Used	Split	Response Type
VQAv2 [12]	214354	2000	val	OE
GQA [18]	12578	2000	testdev	OE
VizWiz [14]	4319	2000	val	OE
TextVQA [57]	5000	2000	val	OE
OCR-VQA [44]	100424	2000	testdev	OE
COCO Caption [37]	202654	2000	val	Caption
Nocaps [2]	4500	2000	val	Caption
Flickr [48]	4999	2000	val	Caption
VCR [76]	26534	2000	val	Two-stage OE
VSR [67]	1221	1221	test	Binary Choice
OK-VQA [43]	5046	2000	val	OE
A-OKVQA [54]	1145	1145	val	Multiple Choice
ScienceQA [41]	2097	2000	val	Multiple Choice
MME [11]	532	532	none	OE
MMBench [39]	4329	2000	none	Multiple Choice
SEED-Bench [25]	171	171	none	Multiple Choice
HalloQuest [63]	604	604	none	OE
MMHalBench [75]	96	96	none	OE

Table 4. Datasets used in evaluation and sampling configuration.

7.5.4. Referring Expression Grounding: MDETR

We integrate MDETR [19] as a referring expression grounding expert. The model takes natural language expressions (e.g., "the man wearing a red hat") paired with input images and outputs bounding boxes aligned to the given phrase. Predictions are filtered through the same refinement step used in detection, yielding JSON summaries and textual descriptions. Referred regions are rendered using the unified visualization modes.

8. Dataset Detail

We evaluate our system on a diverse collection of VQA, captioning, reasoning, and hallucination benchmarks. For datasets containing more than 2000 samples, we construct a fixed subset of 2000 instances selected at random with a seed value of 1 to ensure strict reproducibility across runs. Datasets with fewer samples are fully included. Each dataset is used with its standard evaluation split, and the same subset is shared across all backbones. Table 4 summarizes the datasets, sample counts, and corresponding evaluation partitions.

8.1. Comprehensive Evaluation

8.1.1. Visual Question Answering

We use the standard evaluation protocols defined by each dataset. The five VQA benchmarks differ in both their question styles and their scoring rules:

- **VQAv2**: Evaluated using the official VQA accuracy metric. A predicted answer receives credit according to the agreement with ten human annotations, computed as $\min(\frac{\#\text{ humans that match}}{3}, 1)$.
- **GQA**: Evaluated with exact-match accuracy over a single ground truth answer. The dataset emphasizes compositional and relational reasoning, and no consensus scoring is used.

- **VizWiz**: Evaluated with the same VQA accuracy metric as VQAv2. Many questions are unanswerable due to low-quality images captured by blind users, but the scoring protocol remains identical.
- **TextVQA**: Evaluated using the VQA accuracy protocol. The metric rewards agreement with human responses but focuses on questions requiring optical character recognition and text grounding.
- **OCR-VQA**: Evaluated with exact-match accuracy. Answers are textual strings extracted from book covers or document images, and partial or approximate matching is not applied.

8.1.2. Visual Captioning

We evaluate caption generation on three benchmarks and report CIDEr as the primary metric. CIDEr measures consensus between a generated caption and multiple human references using TF–IDF weighted n-gram similarity. For each caption, n-gram counts (for n from one to four) are converted into TF–IDF vectors, and similarity is computed with cosine distance followed by a brevity penalty. The final CIDEr score is obtained by averaging the per n-gram scores across all sample and reference pairs.

- **COCO Caption**: Contains diverse everyday scenes with multiple human-written captions per image. We compute CIDEr under the Karpathy validation split.
- **Nocaps**: Focuses on open-vocabulary generalization, containing objects unseen in COCO. CIDEr is computed using the official validation evaluation protocol.
- **Flickr30k**: Centers on human-centric and activity-focused descriptions. We report CIDEr using the standard validation split.

8.1.3. Visual Reasoning

We include two benchmarks that evaluate multi-step visual reasoning and justification quality:

- **VCR**: A visual commonsense reasoning dataset where each question requires selecting both an answer and its supporting rationale. We follow the official Q→AR protocol, which evaluates the joint correctness of answer and rationale in a single stage. A prediction is counted as correct only if the model selects the correct answer and the correct rationale simultaneously. The final score reflects the proportion of samples for which both components are correct.
- **VSR**: A binary visual commonsense verification benchmark in which each image–text pair is labeled as either plausible or implausible. Evaluation uses exact-match accuracy over these binary judgments.

8.1.4. Knowledge VQA

These benchmarks require reasoning that combines visual content with external world knowledge or structured scientific information:

- **OK-VQA**: A knowledge-based VQA dataset where many questions cannot be answered from the image alone. Evaluation follows the standard VQA accuracy metric, which measures agreement with ten human annotations.
- **A-OKVQA**: An expanded version of OK-VQA offering both multi-choice and short-answer formats. Multi-choice questions are scored with exact-match accuracy on the selected option, while short-answer questions use the VQA accuracy protocol. The dataset aims to reduce annotation ambiguity and emphasizes grounded knowledge reasoning.
- **ScienceQA**: A multimodal scientific reasoning benchmark containing images, diagrams, and accompanying textual context. Each question is evaluated using exact-match accuracy on the correct option, and many samples require combining visual cues with grade-school science knowledge.

8.1.5. Comprehensive Benchmark

We evaluate broad multimodal capabilities using three comprehensive benchmarks that measure perception, reasoning, and instruction-following quality across diverse tasks:

- **MME**: A multimodal evaluation suite covering fine-grained perception, attribute recognition, OCR, spatial understanding, and commonsense reasoning. Each subtask contributes a fixed number of points, and the final score is computed as the sum across all subtasks following the official evaluation script. Higher scores indicate stronger multimodal grounding and perception fidelity.
- **MMBench**: A large-scale, human-curated benchmark designed to evaluate broad multimodal understanding. Each question is multiple-choice, and scoring uses exact-match accuracy aggregated over the official validation set. The benchmark covers logical reasoning, visual understanding, math, OCR, and world knowledge.
- **SEED-Bench**: A fine-grained multimodal diagnostic benchmark containing diverse perception and reasoning tasks, including compositionality, spatial relations, color, counting, scene classification, and instruction grounding. All tasks are multiple-choice, and performance is reported using exact-match accuracy under the official evaluation protocol.

8.2. Hallucination Evaluation

All hallucination evaluations follow a unified LLM-based judging protocol using `gpt-4o-mini`. For each dataset, the evaluator receives only the inputs required by that benchmark (e.g., image, question, model output, and/or human references) and must assign exactly one categorical label according to the benchmark-specific rubric. All evaluations are conducted in small batches with fixed prompt templates, and final metrics are reported as normalized label frequencies.

HaloQuest. Each HaloQuest sample contains a question, a model-generated answer, and a list of human reference answers (no images are used). For each batch of ten examples, the evaluator receives these fields verbatim and outputs one of *hallucination*, *misperception*, or *not*. The final scores aggregate the proportion of each label both per hallucination type and over the full dataset.

You are a visual reasoning evaluator.
For each example, classify the model's answer into one of the following categories:

- **hallucination**: mentions an object, attribute, or action not supported by the image.
- **misperception**: refers to a real visual concept but with incorrect details.
- **correct**: consistent with the groundtruth description.

Respond with exactly one line per example:
`Example i: hallucination / misperception / not`

MMHalBench. The evaluator receives the image, question, ground-truth answer, and model answer for each sample and assigns one of *correct*, *grounded*, *misperception*, or *hallucination*. Labels reflect whether the prediction factually matches the ground truth, references only visually supported content, misinterprets visible elements, or introduces content absent from the image.

You are an expert multimodal evaluator.
Given an image, a question, a ground-truth answer, and a model's answer, determine the correct label according to the rules below:

- **correct**: matches the ground-truth factually.
- **grounded**: refers only to visible evidence, even if not exactly matching the ground truth.
- **misperception**: describes something visible but with incorrect details.
- **hallucination**: mentions an object or fact not visible in the image.

Respond with exactly **one** label word.

VizWiz-H. We use a two-stage procedure. First, any prediction that matches at least three human reference answers (case-insensitive) is labeled *correct*. All remaining samples are judged by the LLM, which receives the question, model answer, and all reference answers. The evaluator outputs one of *hallucination*, *misperception*, or *grounded*, depending on whether the answer introduces unsupported content, misreads visible information, or stays consistent with the scene while not matching the references.

You are a visual reasoning evaluator.
For each example, classify the model's answer into one of the categories below:

- **hallucination**: mentions an object, attribute, or action that does not exist or is unsupported by the image.
- **misperception**: refers to a real visual concept but with incorrect details (such as wrong color, count, or attribute).
- **grounded**: describes visual content supported by the image, even if not perfectly matching the ground truth.

Respond with exactly one line per example:
`Example i: hallucination / misperception / grounded`

COCO Caption-H. The evaluator receives the model-generated caption and the full set of human reference captions. It outputs one of *hallucination*, *misperception*, *grounded*, or *correct*, depending on whether the caption introduces unsupported content, misstates attributes, provides a partial but accurate description, or matches the references naturally and fully.

You are a caption evaluation assistant.
Your task is to evaluate each model-generated caption against the human reference captions and assign one label:

- **hallucination**: mentions content not supported by any reference.
- **misperception**: describes the same scene but with incorrect attributes, such as color, count, or relations.
- **grounded**: aligns with reference content but is incomplete or lacks detail.
- **correct**: matches the references accurately and naturally.

Respond with exactly one line per example:
`Example i: hallucination / misperception / grounded / correct`

9. Probabilistic Masking

9.1. Top-k Masking

Top-k masking evaluates the LVLM's dependence on the most relevant regions by removing the k highest-scoring components of $r(x \mid q)$ and computing the perturbation metric $U_q^{\text{top-}k} = \mathbb{E}_{\tau \sim r_{\text{top-}k}}[d(f(x), f(x \setminus \tau))]$. In the information-theoretic view, this corresponds to assessing the mutual information $I(Y; X_{\text{high}})$, where X_{high} denotes the feature subset associated with the top-ranked relevance

scores. Because $I(Y; X_{\text{high}})$ quantifies the predictive information that the LVLM extracts from these high-relevance components, large changes in the model output under top-k perturbation indicate that this subset occupies a central role in the information-bearing support of the predictive distribution $p(Y \mid X)$. Top-k masking therefore measures how strongly the model's decision depends on features that the relevance map ranks as most informative.

9.2. Bottom-k Masking

Bottom-k masking suppresses the k lowest-scoring components of $r(x \mid q)$ to evaluate whether the LVLM's prediction remains stable when low-relevance information is removed. The Relevance-Fidelity Index (RFI) captures this behavior as $RFI = 1 - \mathbb{E}_{\tau \sim r_{\text{bottom-}k}}[d(f(x), f(x \setminus \tau))]$. This perturbation regime aligns with the conditional mutual information $I(Y; X_{\text{low}} \mid X_{\text{high}})$, where X_{low} denotes the subset of features with the lowest relevance scores. When $I(Y; X_{\text{low}} \mid X_{\text{high}}) \approx 0$, the high-relevance subset functions as a sufficient representation for predicting Y, meaning the LVLM's predictive distribution can be expressed without dependence on the low-relevance components. In this case, suppressing bottom-ranked regions leaves the output unchanged and results in high RFI values. Bottom-k masking therefore provides an information-theoretic test of whether the relevance map identifies a feature subset that is sufficient for the LVLM's prediction.

9.3. Effect of Mask Count

The number of masks determines how well the perturbation distribution is sampled and therefore directly influences the stability of the expectation $\mathbb{E}_\tau[d(f(x), f(x \setminus \tau))]$. Too few masks lead to high variance in the estimated embedding shift, while excessively many masks yield diminishing gains in stability relative to their computational cost. In practice, we find that using 16 masks provides a reasonable balance: it samples the perturbation space densely enough for the distribution of embedding shifts to exhibit stable central tendency, yet remains efficient enough to integrate into large-scale LVLM evaluations without disproportionate overhead. Although increasing the mask count beyond 16 can further reduce Monte Carlo variance, the improvements are modest compared to the additional computation, and the qualitative behavior of both top-k and bottom-k masking remains unchanged. Thus, $M = 16$ serves as a practical operating point that offers adequate sampling fidelity for assessing perturbation-driven behavior while avoiding unnecessary expansion of compute.

10. Expert Selection Results

Word clouds in Fig. 5 provide a qualitative view of the query terms most associated with each visual expert. Although

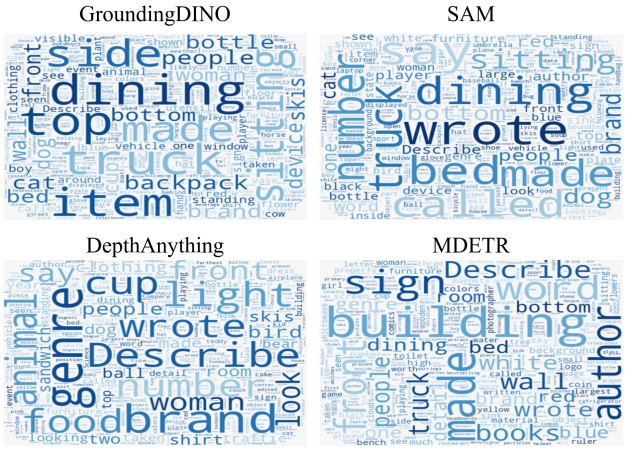

Figure 5. **Expert-specific word distributions.** Word clouds visualize the most frequent query terms or visual concepts attended by each visual expert within the *DnR* framework. GroundingDINO primarily focuses on concrete object-level entities (e.g., "truck," "item," "backpack"), SAM highlights region-based and descriptive terms reflecting segmentation-driven reasoning, DepthAnything emphasizes spatial and depth-related cues (e.g., "light," "floor," "front"), and MDETR captures text–vision correspondence with concept-level words (e.g., "building," "author," "sign"). Together, these distributions reveal how each expert contributes distinct yet complementary perspectives to query-conditioned visual grounding.

these patterns should not be interpreted as strict functional boundaries, several consistent tendencies are observable. GroundingDINO [40] often attends to concrete object nouns, SAM [22] frequently highlights region- or attribute-oriented descriptors, DepthAnything [69] tends to surface terms related to spatial layout, and MDETR [19] appears more sensitive to concept-level or text-aligned phrases. These trends suggest that each expert responds preferentially to different aspects of the input, even though the experiment here is intentionally simple and not designed for strong causal claims.

Importantly, these observations support the feasibility of a more sophisticated policy-driven expert selection network in future work. Even this lightweight analysis indicates that experts expose complementary semantic signatures, implying that a learned policy could exploit such signals at larger scale. While our study limits itself to a minimal selector for clarity, the underlying patterns suggest ample room for expanding toward richer, data-driven expert orchestration.

MME [11]	IDEFICS [24]		InstructBLIP [8]		MiniGPTv2 [77]		LLAVA 1.6 [38]		PaliGemma [9]		CogVLM [17]		Qwen2.5-VL [49]	
	DRAFT	*DnR*	DRAFT	*DnR*	DRAFT	*DnR*	DRAFT	*DnR*	DRAFT	*DnR*	DRAFT	*DnR*	DRAFT	*DnR*
existence	184.21	192.11	184.21	192.11	50	50	200	200	192.11	192.11	200	200	200	200
count	55.26	55.26	65.79	65.79	47.37	47.37	136.84	136.84	168.42	168.42	97.37	97.37	152.63	152.63
position	44.74	47.37	50	50	50	50	105.26	105.26	118.42	118.42	73.68	73.68	144.74	168.42
color	128.95	136.84	136.84	136.84	50	50	168.42	173.68	176.32	176.32	113.16	136.84	184.21	176.32
posters	136.84	136.84	113.16	113.83	57.89	57.89	150	150	78.95	78.95	121.05	121.05	150	150
celebrity	144.74	160.53	81.58	73.68	160.53	173.68	152.63	152.63	73.68	73.68	89.47	97.37	184.21	176.32
scene	121.05	121.05	152.63	152.63	73.68	73.68	165.79	168.42	142.11	142.11	150	165.79	168.42	168.42
landmark	176.32	184.21	152.63	152.63	57.89	65.79	152.63	152.63	157.89	165.79	121.05	121.05	192.11	192.11
artwork	94.74	92.11	73.68	73.68	47.37	63.16	118.42	126.32	102.63	105.26	97.37	97.37	136.84	136.84
OCR	50	50	65.79	65.79	57.89	57.89	65.79	73.68	81.58	81.58	65.79	65.79	136.84	136.84
Perception	1136.84	1176.32	1076.32	1076.32	652.63	689.47	1415.79	1439.47	1292.11	1302.63	1128.95	1176.32	1650	1657.89
commonsense_reasoning	78.95	81.58	73.68	73.68	71.05	63.16	107.89	107.89	65.79	65.79	94.74	86.84	152.63	152.63
numerical_calculation	50	50	44.74	44.74	47.37	44.74	42.11	42.11	15.79	15.79	50	50	128.95	128.95
text_translation	71.05	68.42	50	50	52.63	57.89	73.68	73.68	7.89	7.89	52.63	52.63	184.21	184.21
code_reasoning	55.26	55.26	50	50	55.26	55.26	55.26	57.89	52.63	52.63	57.89	57.89	152.63	152.63
Cognition	255.26	255.26	218.42	218.42	226.32	221.05	278.95	281.58	142.11	142.11	255.26	247.37	618.42	618.42
TOTAL	1392.11	1431.58	1294.74	1295.31	878.95	910.53	1694.74	1721.05	1434.21	1444.74	1384.21	1423.68	2268.42	2276.32

Table 5. MME benchmark results across vision-language models (DRAFT vs *DnR*).

MMBench [39]	IDEFICS [24]		InstructBLIP [8]		MiniGPTv2 [77]		LLAVA 1.6 [38]		PaliGemma [9]		CogVLM [17]		Qwen2.5-VL [49]	
	DRAFT	*DnR*	DRAFT	*DnR*	DRAFT	*DnR*	DRAFT	*DnR*	DRAFT	*DnR*	DRAFT	*DnR*	DRAFT	*DnR*
action_recognition	52.63	52.63	57.89	63.16	15.79	15.79	78.95	78.95	84.21	84.21	89.47	94.74	94.74	94.74
attribute_comparison	36.84	36.84	31.58	36.84	21.05	21.05	63.16	63.16	63.16	68.42	78.95	84.21	78.95	89.47
attribute_recognition	47.37	47.37	47.37	42.11	36.84	36.84	68.42	68.42	63.16	73.68	73.68	73.68	94.74	94.74
celebrity_recognition	73.68	78.95	57.89	57.89	47.37	47.37	100	100	89.47	89.47	100	100	100	100
function_reasoning	42.11	36.84	42.11	31.58	42.11	52.63	84.21	89.47	52.63	52.63	84.21	84.21	94.74	100
future_prediction	47.37	47.37	42.11	42.11	47.37	47.37	68.42	68.42	21.05	26.32	47.37	47.37	68.42	68.42
identity_reasoning	26.32	21.05	47.37	47.37	36.84	36.84	94.74	94.74	84.21	84.21	94.74	94.74	89.47	89.47
image_emotion	47.37	47.37	42.11	47.37	31.58	31.58	68.42	68.42	78.95	84.21	73.68	84.21	78.95	78.95
image_quality	36.84	36.84	57.89	57.89	52.63	63.16	63.16	68.42	94.74	94.74	63.16	63.16	89.47	84.21
image_scene	73.68	73.68	73.68	73.68	42.11	42.11	100	100	73.68	78.95	94.74	94.74	100	100
image_style	78.95	78.95	68.42	68.42	36.84	36.84	89.47	94.74	89.47	89.47	89.47	89.47	100	100
image_topic	68.42	68.42	78.95	78.95	42.11	42.11	94.74	94.74	89.47	89.47	84.21	84.21	94.74	94.74
nature_relation	52.63	52.63	42.11	47.37	47.37	52.63	78.95	78.95	63.16	57.89	84.21	84.21	89.47	94.74
object_localization	31.58	36.84	52.63	52.63	26.32	26.32	57.89	63.16	73.68	73.68	68.42	68.42	78.95	78.95
ocr	63.16	63.16	63.16	68.42	36.84	36.84	89.47	89.47	78.95	78.95	94.74	94.74	94.74	94.74
physical_property_reasoning	52.63	52.63	47.37	47.37	47.37	42.11	73.68	73.68	78.95	78.95	73.68	73.68	78.95	78.95
physical_relation	42.11	42.11	36.84	47.37	42.11	47.37	63.16	73.68	47.37	47.37	52.63	52.63	57.89	52.63
social_relation	47.37	52.63	36.84	36.84	26.32	26.32	84.21	84.21	78.95	73.68	84.21	84.21	89.47	89.47
spatial_relationship	47.37	47.37	36.84	36.84	36.84	36.84	42.11	42.11	42.11	42.11	42.11	42.11	57.89	63.16
structuralized_imagetext_understanding	31.58	31.58	73.68	73.68	36.84	42.11	63.16	63.16	52.63	52.63	63.16	63.16	89.47	89.47
TOTAL	50	50.26	51.84	52.89	37.63	39.21	76.32	77.89	70	71.05	76.84	77.89	86.05	86.84

Table 6. MMBench results across vision-language models (DRAFT vs *DnR*).

SEED-Bench [25]	IDEFICS [24]		InstructBLIP [8]		MiniGPTv2 [77]		LLAVA 1.6 [38]		PaliGemma [9]		CogVLM [17]		Qwen2.5-VL [49]	
	DRAFT	*DnR*	DRAFT	*DnR*	DRAFT	*DnR*	DRAFT	*DnR*	DRAFT	*DnR*	DRAFT	*DnR*	DRAFT	*DnR*
Instance Attributes	47.37	47.37	68.42	68.42	36.84	47.37	84.21	89.47	78.95	78.95	78.95	78.95	94.74	94.74
Instance Interaction	42.11	47.37	63.16	68.42	36.84	42.11	84.21	84.21	73.68	73.68	68.42	68.42	94.74	94.74
Instance Location	42.11	42.11	63.16	63.16	36.84	36.84	78.95	84.21	68.42	68.42	68.42	68.42	84.21	84.21
Text Understanding	42.11	42.11	57.89	57.89	36.84	31.58	78.95	78.95	57.89	63.16	68.42	68.42	84.21	84.21
Instances Counting	31.58	31.58	47.37	52.63	31.58	26.32	73.68	68.42	52.63	57.89	57.89	57.89	84.21	84.21
Spatial Relation	26.32	26.32	47.37	52.63	26.32	26.32	57.89	57.89	52.63	52.63	57.89	57.89	78.95	78.95
Scene Understanding	21.05	21.05	47.37	47.37	26.32	26.32	52.63	52.63	52.63	52.63	52.63	52.63	73.68	78.95
Visual Reasoning	21.05	21.05	36.84	42.11	21.05	26.32	52.63	52.63	42.11	47.37	47.37	52.63	68.42	68.42
Instance Identity	15.79	15.79	31.58	31.58	15.79	21.05	31.58	31.58	42.11	42.11	26.32	26.32	57.89	63.16
Overall Accuracy	32.16	32.75	51.46	53.80	29.82	31.58	66.08	66.67	57.89	59.65	58.48	59.06	80.12	81.29

Table 7. SEED-Bench results across vision-language models (DRAFT vs *DnR*).

HaloQuest [63]		Overall			Insufficient Context			False Premises			Visual Challenge		
	Stage	H	M	C	H	M	C	H	M	C	H	M	C
IDEFICS [24]	Draft	43.34	22.05	34.62	54.17	2.50	43.33	65.35	2.31	32.34	10.50	61.33	28.18
	DnR	40.87	23.76	35.37	54.17	5.00	40.83	62.70	1.65	35.65	6.73	65.14	28.12
InstructBLIP [8]	Draft	33.73	20.48	45.79	42.50	0.00	57.50	46.53	2.31	51.16	12.15	59.12	28.73
	DnR	33.02	21.54	45.44	39.17	0.00	60.83	48.84	1.65	49.50	11.05	62.98	25.97
MiniGPTv2 [77]	Draft	20.22	29.75	50.03	23.33	0.00	76.67	29.04	10.23	60.73	8.29	79.01	12.71
	DnR	19.42	29.52	51.06	20.83	0.00	79.17	29.70	12.87	57.43	7.73	75.69	16.57
LLAVA 1.6 [38]	Draft	26.33	13.96	59.71	37.50	0.00	32.50	37.62	0.99	61.39	3.87	40.88	55.25
	DnR	25.65	14.57	59.79	36.67	0.83	62.50	36.96	1.98	61.06	3.31	40.88	55.80
PaliGemma [9]	Draft	20.67	14.96	64.37	25.00	0.83	74.17	28.71	7.59	63.70	8.29	36.46	55.25
	Refine	16.41	16.63	66.96	22.50	0.83	76.67	25.08	4.29	70.63	1.66	44.75	53.59
CogVLM [17]	Draft	19.24	15.60	65.16	26.67	0.83	72.50	22.77	0.66	76.57	8.29	45.30	46.41
	DnR	17.82	16.72	65.46	25.83	0.00	74.17	23.76	0.66	75.58	3.87	52.49	43.65
Qwen2.5-VL [49]	Draft	3.48	11.90	84.63	2.50	97.50	0.00	4.62	0.33	95.05	3.31	35.36	61.33
	DnR	3.07	12.30	84.63	2.50	0.00	97.50	5.61	0.66	93.74	1.10	37.23	61.67

Table 8. HaloQuest Hallucination Evaluation details across three types.

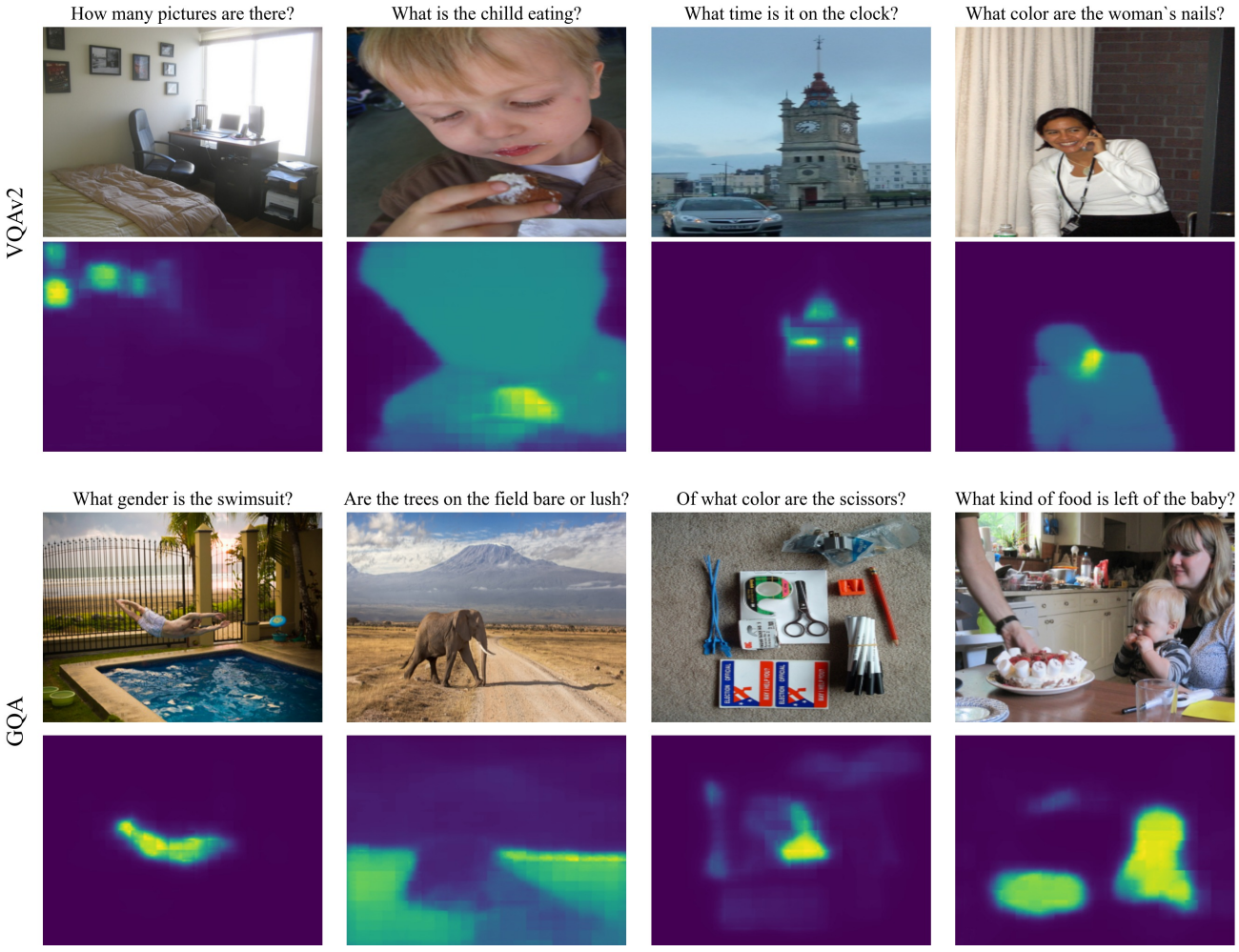

Figure 6. Visualization relevance map $r(x|q)$ of VQAv2, GQA dataset

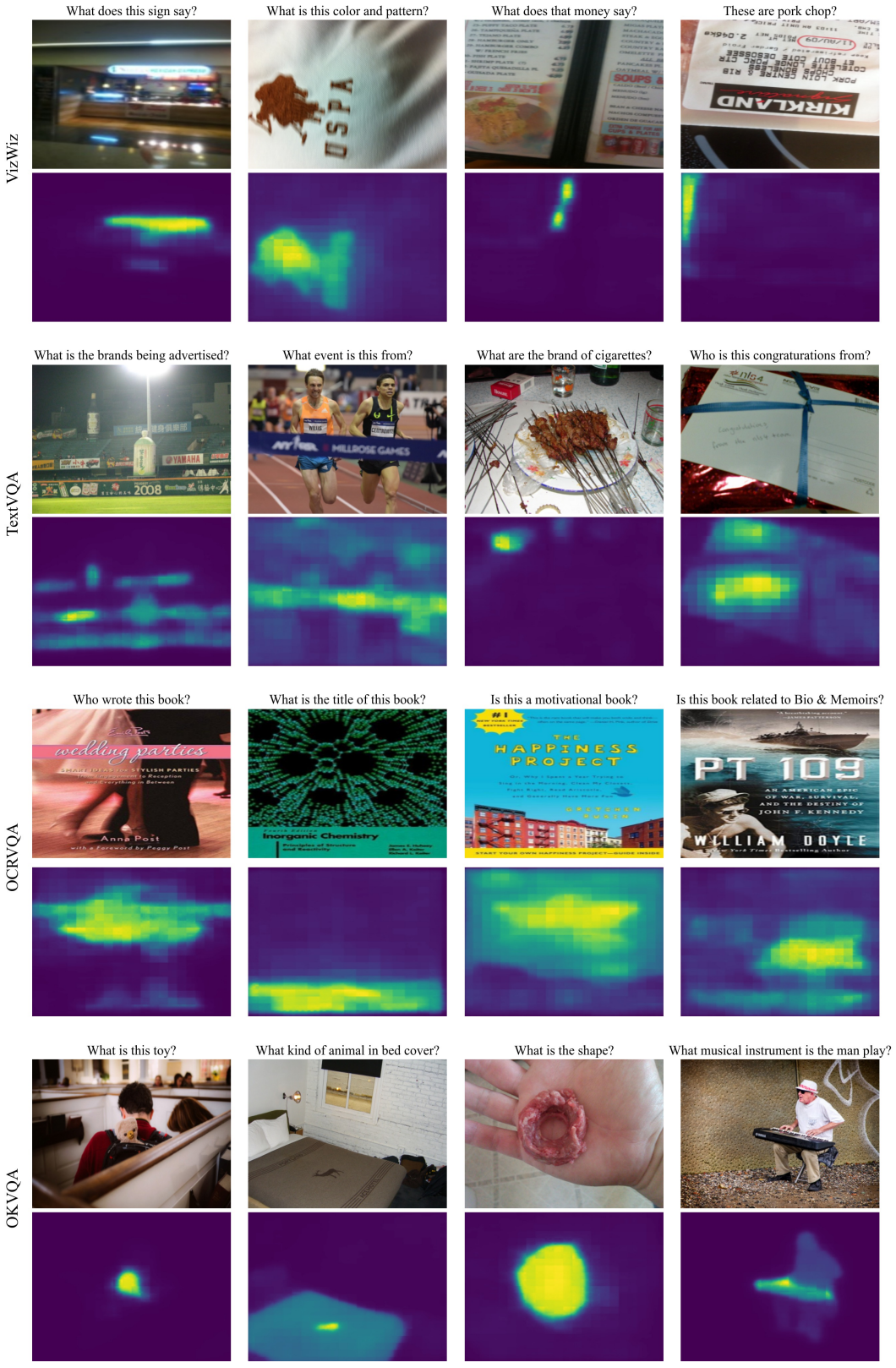

Figure 7. Visualization relevance map $r(x|q)$ of VizWiz, TextVQA, OCRVQA, OKVQA dataset

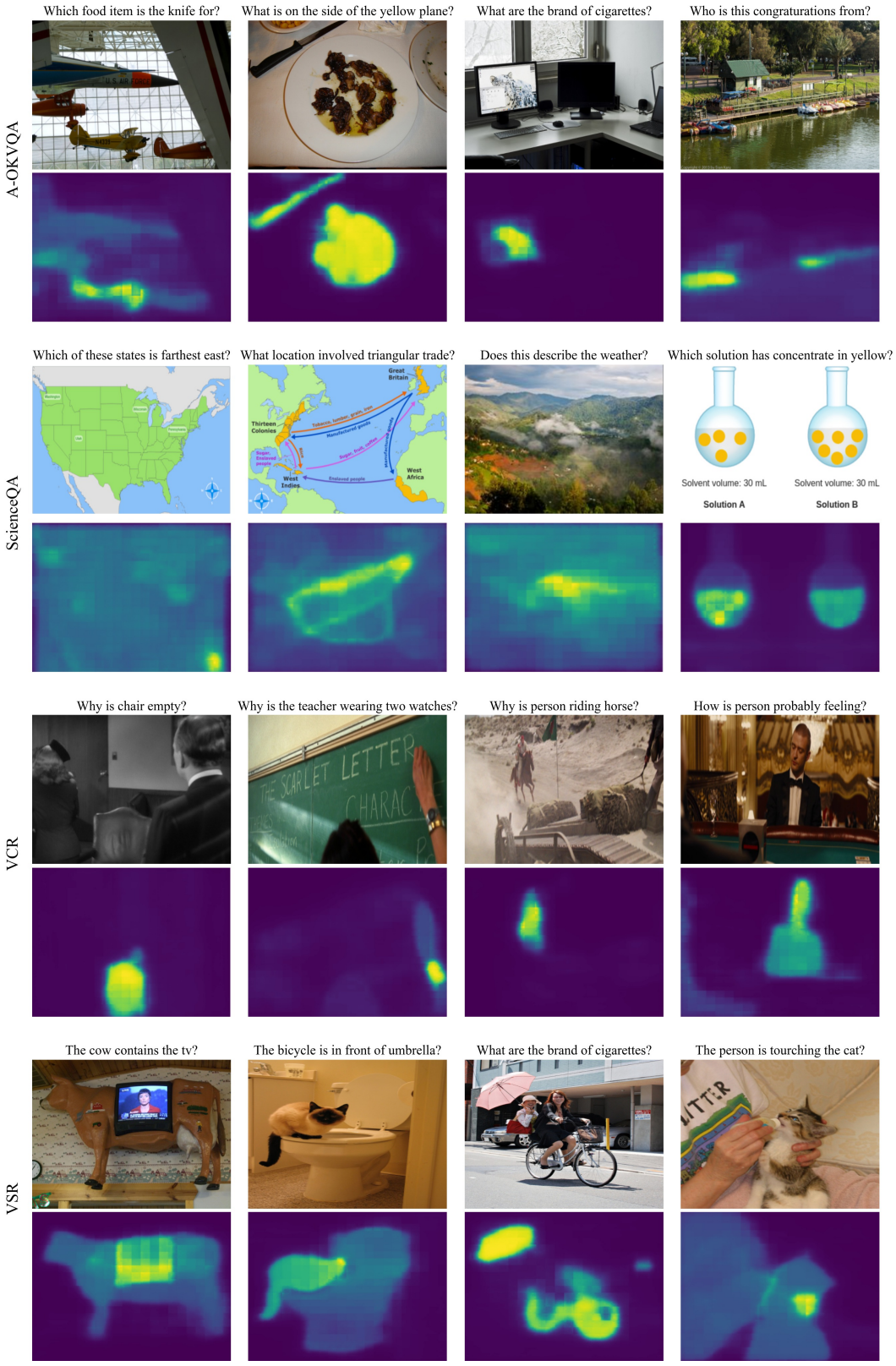

Figure 8. Visualization relevance map $r(x|q)$ of A-OKVQA, ScienceQA, VCR, VSR dataset

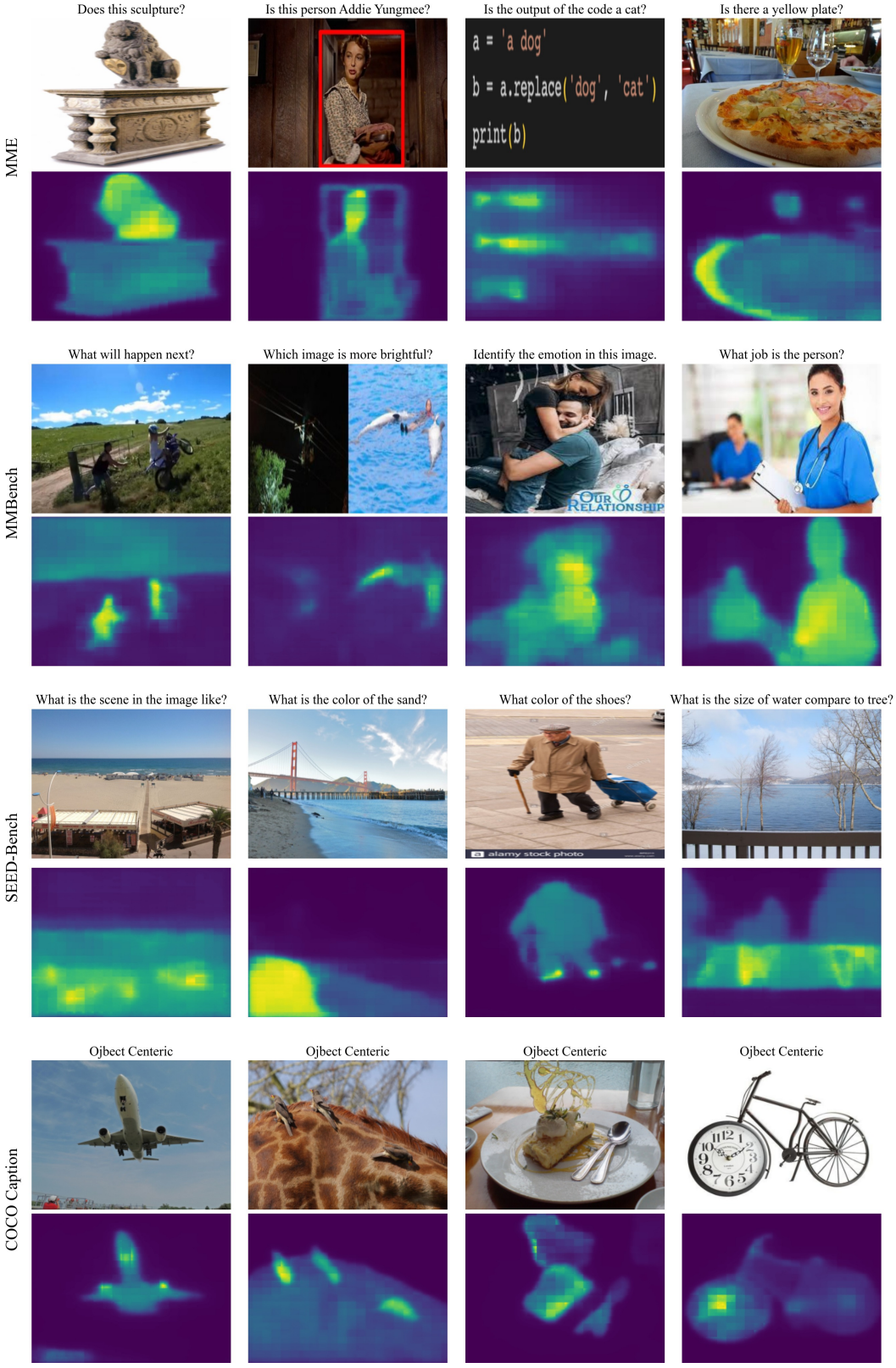

Figure 9. Visualization relevance map $r(x|q)$ of MME, MMBench, SEED-Bench, COCO Caption dataset

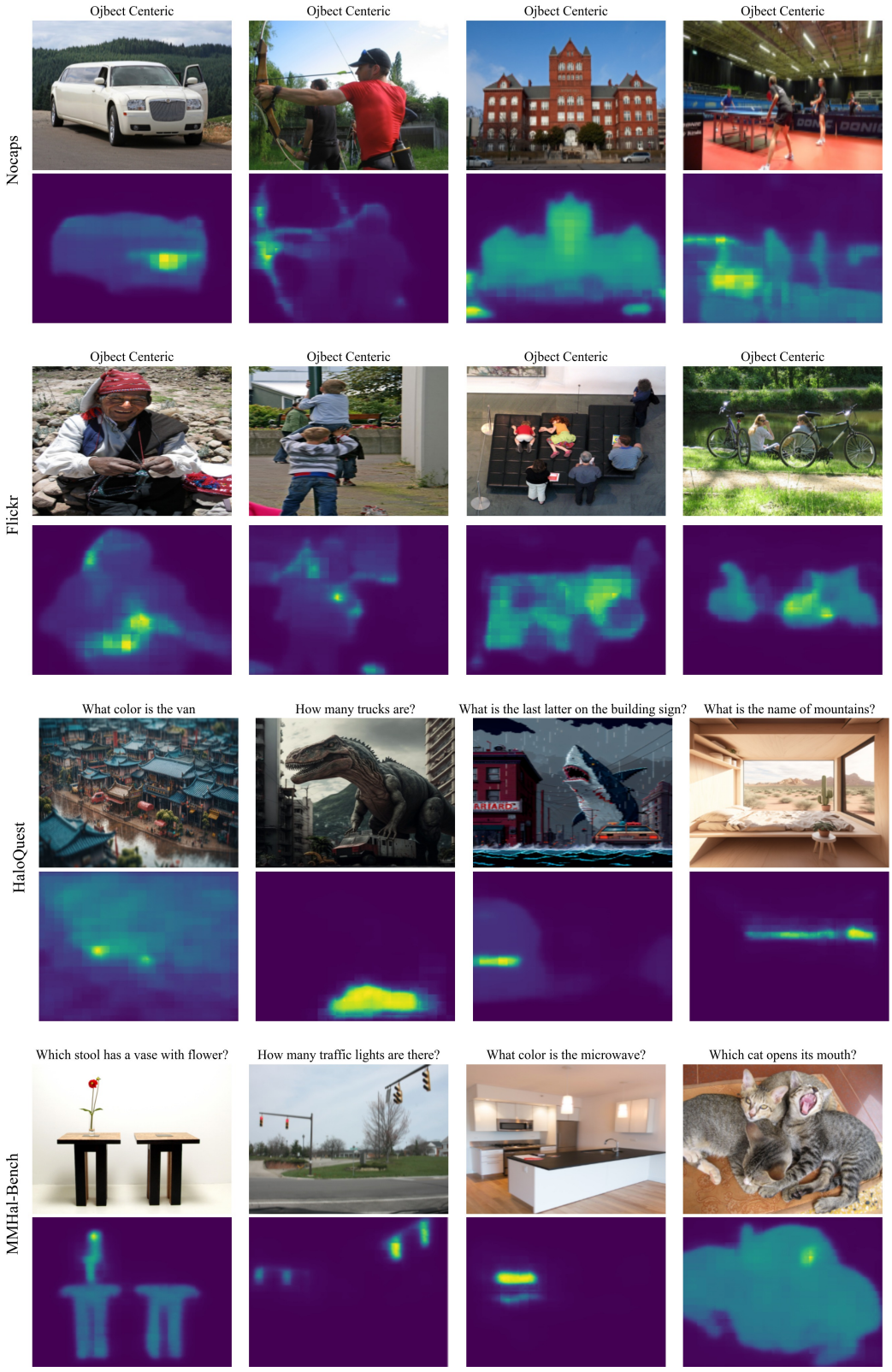

Figure 10. Visualization relevance map $r(x|q)$ of Nocaps, Flickr, Haloquest, MMhal-Bench dataset