Vision-Language Models Struggle to Align Entities across Modalities

Iñigo Alonso[1], Gorka Azkune[2], Ander Salaberria[2], Jeremy Barnes[2], Oier Lopez de Lacalle[2]

[1]School of Informatics, The University of Edinburgh
[2]HiTZ Center - Ixa, University of the Basque Country UPV/EHU
Correspondence: ialonso@ed.ad.uk

Abstract

Cross-modal entity linking refers to the ability to align entities and their attributes across different modalities. While cross-modal entity linking is a fundamental skill needed for real-world applications such as multimodal code generation, fake news detection, or scene understanding, it has not been thoroughly studied in the literature. In this paper, we introduce a new task and benchmark to address this gap. Our benchmark, MATE, consists of 5.5k evaluation instances featuring visual scenes aligned with their textual representations. To evaluate cross-modal entity linking performance, we design a question-answering task that involves retrieving one attribute of an object in one modality based on a unique attribute of that object in another modality. We evaluate state-of-the-art Vision-Language Models (VLMs) and humans on this task, and find that VLMs struggle significantly compared to humans, particularly as the number of objects in the scene increases. Our analysis also shows that, while chain-of-thought prompting can improve VLM performance, models remain far from achieving human-level proficiency. These findings highlight the need for further research in cross-modal entity linking and show that MATE[1] is a strong benchmark to support that progress.

1 Introduction

Several real-world applications demand the ability to perform cross-modal entity linking, i.e., being able to align entities and attributes across modalities. In autonomous driving, for example, a single image of a scene may contain multiple entities, such as pedestrians and other vehicles. Additionally, textual or structured data about these entities, provided by smart devices or other cars, can include information like speed or future trajectory. While some attributes, such as vehicle color or shape, are shared between visual and textual representations, others, like speed, exist only in the text. To navigate effectively, the car must link vehicles in the image to their corresponding textual data, creating a unified representation of the scene. The same is true for other tasks such as multimodal code generation (Mu et al.; Li et al., 2024b), multimodal fake news detection (Jing et al., 2023; Ma et al., 2024) and multimodal scene understanding (Su et al., 2024; Li et al., 2024a).

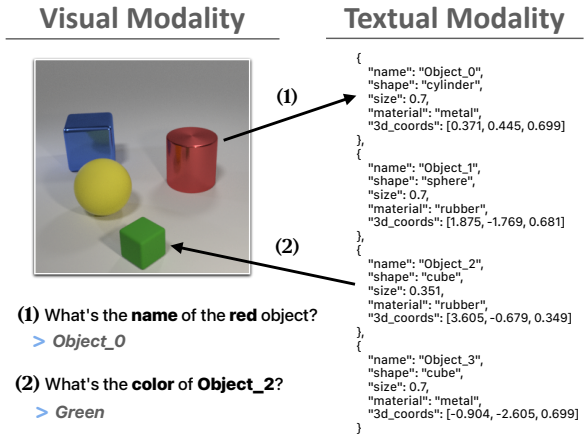


Figure 1: Example of two MATE questions: (1) **Image-to-text**: The model must identify the object in the image based on a visual attribute (red) and retrieve its name from the text ("Object_0"). (2) **Text-to-image**: The model must locate the object in the text using its name and determine its color, which is only present in the image. Both tasks require linking entities across modalities, but in opposite directions.

We can state, thus, that cross-modal entity linking is a basic ability needed to enable further applications of multimodal artificial intelligence systems. However, to the best of our knowledge, no exhaustive and targeted studies can be found in the literature. To fill that gap, in this paper we analyze the capabilities of current Vision-Language Models (VLM) for cross-modal entity linking. Specifically, we build a new multimodal question-answering

[1]Our dataset, evaluation results, and code are publicly available at https://github.com/hitz-zentroa/MATE

Findings of the Association for Computational Linguistics: ACL 2025, pages 18846–18862

benchmark called MATE that contains synthetic images of 3D scenes aligned with textual representations of those scenes (Figure 1). We use synthetic images as this methodology has a large advantage over real-world data – namely, we can control for all the variables of the problem without interference from confounding factors such as object-attribute recognition or visual disambiguation, ensuring that we evaluate cross-modal entity linking independently. The task we use to evaluate the ability of models to perform cross-modal entity linking is shown in Figure 1. Given a pointer attribute which is unique in one of the modalities (e.g., the red color in the first question), we ask for a target attribute of that object which is shown in the other modality (e.g., the name of the referred entity). As the modalities we consider are visual and textual, we have image-to-text (question 1 of Figure 1) and text-to-image tasks (question 2), depending on where the pointer and target attributes exist. We evaluate and analyze open- and closed-weight VLMs on MATE and perform further human evaluations. As a result of our experiments, we find that:

1) VLMs and humans show very different behaviors for cross-modal entity linking: While humans achieve almost perfect performance, current VLMs fail to consistently align attributes across modalities out-of-the-box. Furthermore, VLMs are generally worse for the image-to-text variant; humans show balanced performance. Finally, the performance of VLMs is heavily influenced by the number of objects in the scene and the target attribute requested, whereas human performance is stable across different configurations.

2) The real challenge of our task resides in the cross-modal setting: We show that VLMs can proficiently solve the task of entity-attribute linking when only one modality is considered. However, VLM performance decreases significantly as the number of attributes required to link entities across both modalities increases.

3) MATE is an effective benchmark for evaluating cross-modal entity linking in VLMs: Our results indicate that MATE is a useful resource for evaluating the cross-modal alignment capabilities of current models. While human performance remains consistently high, VLMs suffer a significant drop as scene complexity increases, highlighting the challenges that cross-modal entity linking presents for current models.

2 Related Work

Several multimodal tasks in the literature are related to the one we propose, often grouped under the generic term of *visual grounding*. For example, Referring Expression Comprehension (REC) (Kazemzadeh et al., 2014) requires identifying the image region described by a textual mention, typically referring to objects or physical entities along with their attributes and relations to other objects. Similarly, the Situated and Interactive Multimodal Conversations dataset (SIMMC) (Moon et al., 2020; Kottur et al., 2021) introduces a multimodal dialogue task where a system assists users in a virtual shopping scenario. To complete the task, the system must link visual objects to their textual metadata and search for relevant information. While both tasks share the challenge of aligning visual and textual content, our task extends this further by requiring explicit cross-referencing of object attributes across modalities. In particular, SIMMC avoids this challenge in its shared task by providing gold object IDs, eliminating the need for linking from raw multimodal inputs.

Multimodal Entity Linking (MEL) (Gan et al., 2021; Adjali et al., 2020; Song et al., 2024) is a related task where mentions in multiple modalities are disambiguated by linking them to the corresponding named entities in a knowledge base such as Wikipedia. While previous research (Yao et al., 2024) has primarily focused on scenarios where the image provides supporting visual information for a single entity, our cross-modal entity linking setup involves multiple entities present in both the image and the knowledge base. Similarly to MEL, our task also requires linking textual mentions with visual regions, but we go further by considering pointer attributes of entities (e.g., color and shape) and ensuring that both visual and textual descriptions of the objects are linked to accurately solve the task. This requires the model to perform both visual and textual searches to establish a coherent link across modalities.

All these tasks and many other related ones rely on the core ability of *visual search*, the process of efficiently recognizing and localizing key objects within given scenes, a long-studied topic in cognitive sciences (Peelen and Kastner, 2011; Wolfe, 2020; Wolfe et al., 2011). Several computational models have been proposed for visual search, showing the difficulties of matching human performance (Sclar et al., 2020; Yang et al., 2020; Zhang et al.,

2018; Wu and Xie, 2024). Recent work (Campbell et al., 2024) proposes that those difficulties are closely related to the binding problem (Roskies, 1999), i.e., the ability to associate one feature of an object (e.g., its color) with the other features of that object (e.g., its shape and location).

Our task is also based on visual search, but it adds the homologous textual search and poses the challenge of *linking* the textual description of an object (with a given set of attributes) with its visual description using a unique pointer attribute (Section 3). To the best of our knowledge, no similar task has been studied previously.

3 Cross-modal Entity Linking

Cross-modal entity linking refers to the ability to understand that an entity described in two different modalities is actually the same entity. For example, in Figure 1, the red cylinder is represented in two ways: visually, with attributes such as color, shape, and size, and textually, as a set of attribute-value pairs.

In order to evaluate whether VLMs possess this ability, we propose a question-answering task. We create 3D scenes containing multiple objects and provide a textual collection of attribute-value pairs for all objects in the scene (see Figure 1). Importantly, while some attributes are shared across modalities, others are exclusively available in one modality. In Figure 1, the color attribute is not included in the textual modality, while the object's name appears only in text and other attributes (shape, size, and material) are common to both.

Using this setup, we design questions that ask for the value of a particular object's attribute. In Figure 1, the first question asks for the name of the red object. In this example, the color red acts as the *pointer attribute*, which identifies the object in the scene. To avoid ambiguity, each pointer attribute is unique to a single object. The "name" attribute, only available in the textual modality, serves as the *target attribute*. Answering correctly requires linking the object's visual and textual representations.

Question 1 in Figure 1 is an *image-to-text* task because the pointer attribute is only represented in the visual modality and the target attribute in the text. Question 2, instead, is *text-to-image*, where the pointer attribute "name" is only represented in text, and the target attribute is the color, represented only in the image. This question-answering task is a suitable proxy for testing the ability of cross-modal entity linking, since to achieve high performance, models must link two different representations of the same entity.

3.1 The MATE Benchmark

To perform this evaluation, we introduce MATE, a benchmark dataset consisting of 5,500 question-answering examples. Each example features a scene composed of three to ten 3D geometric objects with various colors, shapes, materials, and sizes (see Figure 1 for reference). Each scene is represented in both the visual modality (image) and the textual one as a list of objects and their attributes (shown in JSON format in Figure 1). The scenes in MATE are based on the CLEVR dataset (Johnson et al., 2017), but we extend them with additional shapes and uniquely identifiable object names.

MATE includes one question per example, and each question features a pointer and a target attribute. When the pointer or target attribute belongs to the visual modality, we use *color* or *shape*. For attributes residing in the textual modality, we use *name*, *rotation*, *size*, and *3D coordinates*. Additionally, the dataset features a *material* attribute, which, although not used as a pointer or target due to its limited value range, still serves as a descriptive property (see Appendix A for a list of all attributes). Note that even though every serialized scene in our dataset contains all these attributes, the scene included in the prompt never contains the attribute pointed to or retrieved from the image. This prevents models from relying solely on a single modality. For example, in Figure 1, because the color attribute acts as the pointer attribute in the image-to-text question and as a target attribute in the text-to-image question, it is never included in the serialization provided to the model.

To ensure unbiased evaluation, MATE maintains a balanced distribution of features. The dataset examples are uniformly distributed across the number of objects in the scene. It also provides equal numbers of examples for both image-to-text and text-to-image tasks, with a total of 2,750 examples per setting. Furthermore, pointer-target attribute pairs are uniformly distributed across all object counts, resulting in 43 ± 1.5 examples per attribute pair, object count, and setting. This balanced design makes MATE a robust benchmark for evaluating cross-modal entity linking (see Appendix B for all information included in the dataset).

	Model	Img2Txt	Txt2Img	Avg.
	Human	97.9	97.9	97.9
	Random	25.4	18.5	22.0
Open	LLaVA 1.5	29.3	35.7	32.5
Open	LLaVA 1.6	48.7	61.6	55.2
Open	Molmo	18.1	20.9	19.5
Open	Llama 3.2	37.4	11.4	24.4
Open	Qwen2-VL	72.1	77.2	74.7
Open	Qwen2.5-VL	**75.7**	**84.5**	**80.1**
Closed	Gemini 1.5	63.2	71.2	67.2
Closed	GPT-4o	76.4	79.1	77.8
Closed	Claude 3.5	**80.9**	**85.7**	**83.3**

Table 1: Results of open and closed VLMs in our task. All results are obtained using two-shot prompting. Exact match accuracy is provided for image-to-text (Img2Txt column) and text-to-image (Txt2Img column) configurations. Human and random accuracies are shown as reference.

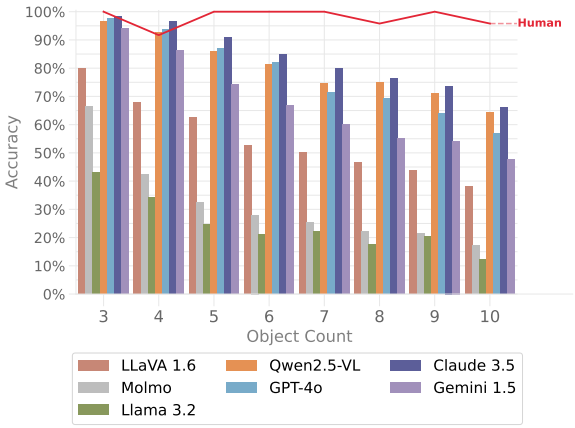


Figure 2: Average accuracy of VLMs and humans for cross-modal entity linking, depending on the number of objects in the scene. VLM performance decreases with the number of objects, whereas humans perform very similarly for all scenarios.

4 Experiments

We conducted our experiments using a variety of open and closed models (see Appendix D for a full list). For open models, we report results for the top-performing model from each family. The families we considered are LLaVA-1.5 (Liu et al., 2024b), LLaVA-1.6 (Liu et al., 2024a), Molmo (Deitke et al., 2024), Llama 3.2 (Dubey et al., 2024), and Qwen2.5-VL (Bai et al., 2025).[2] For closed models, we included GPT-4o, Claude 3.5 and Gemini in our evaluation.[3] All open-weight models where evaluated in a computing cluster using four NVIDIA A100 80GB GPUs. The entire set of experiments was estimated to have required approximately 300 GPU hours. For the closed-weight commercial models, we used their respective official APIs.

We tested all models using zero-, one-, and two-shot prompting. As most evaluated models restrict visual input to a single image, we used one image along with two question-answer examples for two-shot prompting. When the target attribute resided in the visual modality, we included a set of possible options to help the models align their responses with the expected terminology (see Appendix H.2).

We represented the scene in the textual modality using JSON format. In an initial comparative study, we explored how different textual representation formats (JSON, YAML, XML, or verbalized descriptions) affected performance. The results showed no statistically significant differences between formats, though JSON and verbalized representations slightly outperformed the others. Thus, all experiments in this work were conducted using JSON as the scenes' representation format in the textual modality.

4.1 Human Evaluation

We furthermore conducted a human evaluation to quantify the extent to which humans are capable of performing the kind of entity-attribute correlations we expect from VLMs, which serves as an upper-bound for the task.

For this purpose, we selected a subset of 384 examples from our benchmark, ensuring a representative distribution of features.[4] The evaluation examples were distributed among 5 participants (see Appendix C for further details). All of the evaluators were presented with the same image and textual prompts as those used in the VLM evaluations.

[2]These VLMs can be paired with different LLM backbones; we only present results for the best-performing variants: LLaVA 1.5 13B, LLaVA 1.6 34B with Nous-Hermes-2-Yi-34B LLM backbone, Molmo-7B-O-0924, Llama-3.2-11B, Qwen2-VL-7B, Qwen2.5-VL-7B

[3]Tested versions: *gpt-4o-2024-11-20*, *claude-3-5-sonnet-20241022* and *gemini-1.5-flash*

[4]This subset included at least one combination of every target and pointer attribute across examples with all possible object counts from both image-to-text and text-to-image variants.

	Model	Txt2Img		Img2Txt				OSE ↓
		Shape	Color	Name	Coords	Rotation	Size	
	Human	100.0	93.2	96.9	96.9	96.9	100.0	0.0
	Random	24.8	12.2	17.4	18.4	18.4	47.6	N/A
Open	LLaVA 1.5	47.5	24.0	27.8	13.7	24.4	51.5	18.1
	LLaVA 1.6	70.4	53.0	56.0	35.3	37.1	66.5	6.5
	Llama 3.2	7.0	15.7	49.5	15.5	34.4	50.2	83.5
	Molmo	31.4	10.5	21.8	32.9	4.8	12.9	47.1
	Qwen2-VL	86.8	67.8	76.9	62.4	63.9	85.2	**2.6**
	Qwen2.5-VL	**91.7**	**77.4**	**81.5**	**64.8**	**69.9**	**86.7**	7.0
Closed	Gemini 1.5	74.6	67.9	57.7	53.4	60.2	81.8	**2.1**
	GPT-4o	82.2	76.0	74.0	72.2	73.0	86.4	4.4
	Claude 3.5	**90.6**	**80.3**	**78.8**	**78.4**	**75.1**	**91.2**	3.6

Table 2: Accuracy of VLMs for the text-to-image and image-to-text tasks, depending on the target attribute. Human and random performance are also depicted as reference. OSE, Out-of-Scene Error: the percentage of erroneous predictions that do not match any existing target attribute in the scene. Lower scores indicate that the model's errors are more likely due to entity linking mistakes rather than copying or hallucination issues.

5 Results

We report the exact match accuracy across all experiments, as they involved either extractive question-answering (for textual target attributes) or multiple-choice question-answering (for visual target attributes). We also report results for the two-shot settings due to their better performance and stability across inferences. We consider this setting a better representation of the capabilities of these models in this task (see Appendix F).

Table 1 presents the overall results of the VLMs on our MATE benchmark, divided into image-to-text and text-to-image variants, as well as their average. While the task is straightforward for humans (approaching 100% accuracy), it remains challenging for current VLMs, although their performance is well above random chance. The best-performing open-weight VLM falls 17.8 absolute points behind human accuracy, highlighting significant room for improvement. Even the highest-performing closed-weight commercial VLM falls short of human performance by 14.6 absolute points. This gap is particularly notable given the expectation of near-perfect accuracy on such a fundamental task. Therefore, we can conclude that current VLMs, unlike humans, struggle to consistently link representations across different modalities for the same set of objects, which may limit the capacities of those models for more advanced tasks where this ability is required.

Nevertheless, the results reveal a clear progression in VLM capabilities. There is a noticeable trend between model performance, parameter count, and release date, with more recent and larger models consistently outperforming their counterparts (see complete results in Appendix D).

It is also interesting to note that the text-to-image configuration is easier for all VLMs,[5] indicating that it is easier to identify the pointer attribute in the text and link it with the visual attribute of the image. This is not the case of humans, who have the same performance for both task configurations.

In Figure 2 we further analyze the performance of VLMs and humans as the number of objects in the scene increases (3-10). The performance is calculated as the average between image-to-text and text-to-image accuracies. When correlating attributes and linking entities correctly, the number of objects in a scene does increase the difficulty of the task, but humans show no degradation in their performance. On the contrary, for VLMs, performance degradation was significant, even for top-performing models like Claude, which is almost 30 absolute points behind humans for scenes of 10 obejcts. This behavior can be explained by *feature interference*, which tends to increase with

[5]The only exception is Llama3.2. We ignore the reasons behind this phenomenon, but we speculate it could be related to model architecture (the use of cross-attention layers to connect the visual encoder and LLM decoder) and training recipe (the LLM decoder is kept frozen during multimodal training).

the number of objects. According to (Campbell et al., 2024), humans are more robust to feature interference than current VLMs, as observed again in our experiments.

Finally, Table 2 shows the accuracies obtained by VLMs for each target attribute. In theory, once the object has been linked across modalities, copying an attribute as the answer should be equally straightforward for all attributes, especially in the image-to-text setting. However, our results indicate that VLM performance decreases as the range of possible values for the target attribute increases. Moreover, the low percentage of Out-of-Scene Errors reported in Table 2 (percentage of erroneous predictions that do not match any existing target attribute value in the scene), suggests that incorrect predictions still correspond to other objects in the scene. This indicates that these errors are more likely caused by entity linking issues rather than copying or hallucination errors. The exceptions are Llama 3.2 and Molmo, with very high OSE, which accounts for their tendency to hallucinate attribute values.

6 Analysis

Our human evaluations revealed that solving the cross-modal entity linking task typically involves a series of sequential steps. We use the image-to-text scenario in Figure 1 (question 1) as an example:

1) Visual search: The pointer attribute (the red color) is used to identify the object in the image (the red cylinder).

2) Linking attribute identification: Attributes other than the pointer attribute, which distinguish the object from the others in the scene, are identified. In question 1 of Figure 1, the most distinctive linking attribute is the object's cylinder shape, as the red object is the only cylinder in the scene.[6]

3) Textual search: The linking attributes identified in the previous step are used to locate the object in the textual modality. For instance, the object with the attribute *"shape": "cylinder"* is found in the textual data, and its corresponding target attribute, *"name"*, is retrieved.

For the text-to-image setting, the same steps are followed, but in reverse order. In question 2 of Figure 1, for instance, step 2 would involve combining at least two attributes, such as shape and material, or shape and size. In our analysis, we break down the cross-modal entity linking task into these three subtasks to analyze VLM performance and identify the most challenging aspects for these models.

6.1 Visual and textual search

In this section, we evaluate whether VLMs can perform entity-attribute alignment within a single modality (steps 1 and 3 as identified by human annotators). This helps determine whether the poor multimodal results stem from a general failure to extract attributes within the same modality or from the challenges of cross-modal reasoning itself, i.e., linking attribute identification (step 2).

We conduct question-answering tasks within the same modality for both the pointer and target attributes, which we call image-to-image (where both attributes exist in the visual modality) and text-to-text (where both attributes exist in the textual modality).[7] Following the terminology used by Campbell et al. (2024), we are evaluating disjunctive visual and textual searching capabilities separately.

For each case, models were prompted with the task's objective and provided two solved questions as a two-shot prompting setting. For image-to-image tasks, we included a set of possible answer options in order to help the models align their responses with the terminology of hidden attributes (see Appendix H.1 for examples of these unimodal prompts).

Table 3 reports the results of the unimodal search experiments (image-to-image and text-to-text). All models show significantly stronger performance compared to the cross-modal entity linking (Table 1). In particular, Qwen2-VL, Qwen2.5-VL, and the closed-weights commercial VLMs achieve near-perfect accuracy, although a number of VLMs still struggle with entity search. In a number of cases, we observe certain performance discrepancies between image-to-image and text-to-text tasks. We hypothesize that, despite having strong visual capabilities, the LLM backbones of these VLMs play a critical role in determining overall performance, and thus text-to-text performance tends to be higher.

[6] Our benchmark dataset includes all possible linking attribute combinations that uniquely identify the target entity in each example.

[7] Due to implementation issues, the tested VLMs require an input image even for the text-to-text experiments. We ran all the experiments twice, using white images and black images as inputs and found that the performance did not vary. Therefore, we report results using a white image for the text-to-text task.

	Model	Img2Img	Txt2Txt	Avg.
	Human	100.0	99.0	99.5
	Random	18.7	25.3	22.0
Open	LLaVA 1.5	55.3	88.6	72.0
Open	LLaVA 1.6	84.4	98.0	91.2
Open	Molmo	86.9	54.7	70.8
Open	Llama 3.2	68.9	97.1	83.0
Open	Qwen2-VL	**99.7**	98.1	98.9
Open	Qwen2.5-VL	**99.7**	**99.4**	**99.5**
Closed	Gemini 1.5	95.9	**100.0**	98.0
Closed	GPT-4o	**98.4**	**100.0**	**99.2**
Closed	Claude 3.5	97.3	**100.0**	98.7

Table 3: Results of open and closed VLMs in the unimodal variants of our task. All results are obtained using two-shot prompting. Exact match accuracy is provided for image-to-image (Img2Img column) and text-to-text (Txt2Txt column) configurations. Human and random accuracies are shown as reference.

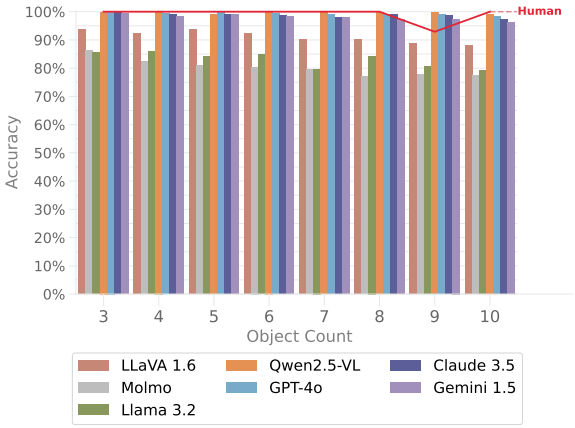


Figure 3: Average accuracy of VLMs for text-to-text and image-to-image tasks depending on the number of objects in the scene. VLMs performance keeps stable regardless of the number of objects, just as humans.

In Figure 5 (see Appendix E), we plot the performance of VLMs for different numbers of objects in the scenes, reporting the average accuracies for both image-to-image and text-to-text tasks. The results show that model performance is much more stable than in the cross-modal scenario (Figure 2), indicating that the number of objects is not a significant factor when only a single modality is considered. These findings align with (Campbell et al., 2024), whose results on disjunctive visual search tasks show that the number of objects does not affect VLM performance when feature interference is low.

Qwen2.5-VL performs near perfectly in unimodal tasks; however, it struggles with cross-modal ones (Table 1), suggesting that its difficulties in MATE stem from the second step: linking attribute identification, making it a perfect candidate for further analysis. In Section 6.2, we will examine the performance of this model in greater depth.

6.2 Linking attribute identification

In the cross-modal setting, once an object is identified in the pointer modality, it must be cross-referenced with objects in the other modality using attributes that distinguish it from others in the scene. In some cases, a single distinctive attribute is sufficient, while in others it is necessary to combine up to three attributes to uniquely identify the object. We refer to these attributes as linking attributes.

In Figure 4 (a), we analyze how the number of required linking attributes influences the model's performance. We measure Qwen2.5-VL's accuracy, the top-performing open-weight model, on examples where the target entity can be traced with a minimum of one, a combination of two, or three attributes, in addition to the 3D coordinates. Although the 3D coordinates are unique to each entity, using them to correctly identify an object is more complicated than combining other attributes. This can be observed in the last column of the figure, labeled *"3dc"*, where we plot the accuracy for examples where the target object can only be linked using the 3D coordinates.

Given the significant impact of object count on performance (see Figure 2), we controlled for this variable by analyzing only scenes with seven objects, as these scenes share an equal distribution of examples requiring one, two, three, or only 3D coordinate linking attributes.

This analysis demonstrates that model performance improves when fewer linking attributes are required, suggesting that the model benefits from a smaller set of distinctive attributes. Furthermore, the model does not appear to simply aggregate all available attributes and directly match the target object. In fact, the results indicate that it behaves more like humans, as the accuracy drops noticeably when additional linking attributes are required. If the model used all attributes in a brute-force manner, we would not see this decline in performance.

To shed more light on which type of attribute is most effective as a linking attribute, we measure the model's accuracy for examples that can be

linked using a single attribute and separate it by attribute types. These results can be seen in Figure 4 (b). These attribute types contribute proportionally when combined in pairs or triplets (results for all possible combinations are provided in Appendix G).

These results highlight a clear underperformance of the model in cases where 3D coordinates are the only linking attribute that uniquely identifies the target object. In these cases, the model continues to rely on attribute overlap to make predictions, with a clear preference towards objects with more shared attributes with the target object rather than uniquely relying on the interpretation of 3D coordinates (see Appendix E). However, we still see that the model has a partial understanding of 3D coordinate interpretation. In cases where the predicted object matched all attributes except for the 3D coordinates, the model was able to predict the correct object in 63% of these.

There are still 14.8% of cases where the predicted object also contained a mismatch in one or more non-coordinate attributes, indicating that the failure was not uniquely due to misinterpreting spatial information but also involved other attribute-level errors. Even in these cases, the model selected an object with a lower Euclidean distance to the gold object in 70% of instances, compared to the average distance of objects with the same number of overlapping attributes.

6.3 Chain-of-Thought Evaluation

The main experiments require models to solve the problem in a single step, although humans intuitively approach this task as a series of logical steps. For complex problems, however, VLMs and Large Language Models (LLMs) generally benefit from breaking a task into a sequence of simpler subtasks (Wei et al., 2022; Chen et al., 2024). This process, known as Chain-of-Thought (CoT), helps guide the final output toward the correct solution of a complex problem.

In order to verify whether poor model performance was due to an inability to handle the task in one step, we conduct a further evaluation of the same task using CoT prompting techniques. Here we prompt the models first to identify the object in the pointer modality that matches the given pointer attribute. Next, they must list this object's attributes, identify those that distinguish it from other objects in the pointer modality (linking attributes), locate the corresponding object in the target modality using these linking attributes, and finally return its target attribute. We conduct this experiment in a two-shot setting (see Appendix H.3 for the complete prompt).

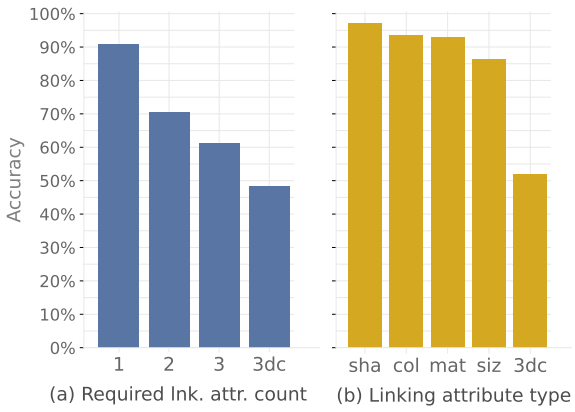


Figure 4: (a) Accuracy of Qwen2.5-VL for examples where the correct object must be identified using one, a combination of two, or three visual attributes (i.e., attributes that can be perceived visually). These attributes are considered in addition to 3D coordinates, which could always uniquely identify objects across modalities but models struggle to interpret. This is evident from the accuracy of the examples that rely solely on 3D coordinates (denoted as *3dc*). (b) Accuracy for examples where the correct object can be linked with just one attribute of a single specific type. For example, in the case of material (*mat*), when the pointed object is the only object in the scene made of rubber or metal. *mat*: material, *sha*: *shape*, *col*: color, *siz*: size, *3dc*: 3D coordinates.

Table 4 presents the overall results along with their differences compared to the scores without CoT. While CoT significantly improves performance for Molmo and Llama 3.2, it only marginally benefits the commercial models and reduces performance in the LLaVA family. Although some models' performance seems strong overall, especially commercial models, all models still experience performance declines as the number of objects in the scene increases, as shown in the column on the right. This indicates that the main issue still persists and is related to VLMs' inability to link entities across modalities, rather than the multi-step nature of the task. Consequently, improving VLMs for cross-modal entity linking will likely require approaches beyond CoT.

6.4 Self-Reflection Evaluation

To evaluate whether self-reflective techniques would tackle this problem better, we evaluated the open-weight self-reflecting visual language model

	Model	All	10 Obj.
	Human	97.9	95.8
	Random	22.0	18.6
Open	LLaVA 1.5	28.1 (-4.5)	18.5 (-6.5)
Open	LLaVA 1.6	40.0 (-15.2)	29.7 (-8.4)
Open	Molmo	45.1 (+25.5)	31.5 (+19.0)
Open	Llama 3.2	53.7 (+29.2)	36.3 (+24.1)
Open	Qwen2-VL	72.9 (-1.8)	54.2 (+0.3)
Open	Qwen2.5-VL	**78.9** (-1.2)	**62.8** (-1.5)
Closed	Gemini 1.5	72.5 (+5.1)	53.2 (+5.5)
Closed	GPT-4o	82.8 (+5.0)	64.6 (+7.7)
Closed	Claude 3.5	**86.2** (+2.9)	**70.5** (+4.5)

Table 4: Results of open and closed VLMs on our task for two-shot CoT prompting. Exact match accuracy is reported for all examples (image-to-text and text-to-image), along with their differences compared to the scores without CoT (originally shown in the Avg. column of Table 1). Results for scenes with 10 objects are also provided.

VL-Rethinker-7B (Wang et al., 2025) using the same 2 shot prompt as models in Table 1. VL-Rethinker-7B achieves 75.7% and 83.4% accuracy in the image-to-data and data-to-image settings, respectively. In comparison, Qwen2.5-VL—its base, non-self-reflective counterpart, achieves 75.7 and 84.5 points in the same settings. This suggests that self-reflecting does not provide any added value for cross-modal entity linking.

7 Conclusions

In this work we demonstrate that VLMs are unable to consistently match the same entity across modalities and retrieve one of its attributes, even guiding their inference with chain-of-thought prompting. To support the evaluation of this fundamental skill, we present MATE, a benchmark designed to assess the proficiency of both current and future models in this area.

Future work could focus on evaluating tasks with greater complexity regarding the pointer and target attributes. One of the main advantages of this benchmark is that it can be easily extended to scenarios involving multiple pointer and target attributes (e.g., *"Identify the name and rotation of the blue sphere"*) and/or a higher number of objects. The last alternative may be especially interesting to evaluate test-time scaling techniques, trying to better align with humans' capability of using more time for more complex scenarios.

Finally, it would be interesting to test models on cross-modal linking in a real-world scenario. One could choose a set of real-world entities (people, cars, etc.) and discretise a set of attributes that uniquely identify the entities (clothing, colour, size, etc). However, the main challenge for such an undertaking is to design the accompanying knowledge base such that you maintain a well-defined scope for the controlled evaluation of cross-modal entity linking.

Limitations

MATE is composed of synthetic images, limiting the variety of phenomena that can occur in natural images. We decided to use synthetic images to control for all the important variables of the problem, but we acknowledge that it would be interesting to build a similar dataset with natural images.

The textual part of our benchmark is provided only in English. This decision is based on the fact that current VLMs are mostly trained with English texts. However, as the multilingual capabilities of such models improve, having multilingual versions of our dataset could offer interesting insights.

Finally, to focus on the cross-modal aligning part, we kept the visual and textual searches simple, using only one pointer attribute. The extension of our benchmark to require combinations of attributes as pointers is straightforward, and could make the task even more difficult.

Acknowledgments

This work is partially supported by the Ministry of Science and Innovation of the Spanish Government (AWARE project TED2021-131617B-I00, DeepKnowledge project PID2021-127777OB-C21), project funded by MCIN/AEI/10.13039/501100011033 and by FEDER, the Basque Government (IXA excellence research group IT1570-22), the European Union under Horizon Europe (Project LUMINOUS, grant number 101135724), and the UK Engineering and Physical Sciences Research Council (grant EP/W002876/1).

References

Omar Adjali, Romaric Besançon, Olivier Ferret, Hervé Le Borgne, and Brigitte Grau. 2020. Multimodal entity linking for tweets. In *European Conference on Information Retrieval*, pages 463–478. Springer.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. 2025. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Declan Iain Campbell, Sunayana Rane, Tyler Giallanza, C. Nicolò De Sabbata, Kia Ghods, Amogh Joshi, Alexander Ku, Steven M Frankland, Thomas L. Griffiths, Jonathan D. Cohen, and Taylor Whittington Webb. 2024. Understanding the limits of vision language models through the lens of the binding problem. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Yangyi Chen, Karan Sikka, Michael Cogswell, Heng Ji, and Ajay Divakaran. 2024. Measuring and improving chain-of-thought reasoning in vision-language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 192–210, Mexico City, Mexico. Association for Computational Linguistics.

Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. 2024. Molmo and pixmo: Open weights and open data for state-of-the-art multimodal models. *CoRR*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Jingru Gan, Jinchang Luo, Haiwei Wang, Shuhui Wang, Wei He, and Qingming Huang. 2021. Multimodal entity linking: a new dataset and a baseline. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 993–1001.

Jing Jing, Hongchen Wu, Jie Sun, Xiaochang Fang, and Huaxiang Zhang. 2023. Multimodal fake news detection via progressive fusion networks. *Information processing & management*, 60(1):103120.

Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. 2017. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2901–2910.

Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. 2014. Referitgame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 787–798.

Satwik Kottur, Seungwhan Moon, Alborz Geramifard, and Babak Damavandi. 2021. SIMMC 2.0: A task-oriented dialog dataset for immersive multimodal conversations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4903–4912, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jianxin Li, Guannan Si, Pengxin Tian, Zhaoliang An, and Fengyu Zhou. 2024a. Overview of indoor scene recognition and representation methods based on multimodal knowledge graphs. *Applied Intelligence*, 54(1):899–923.

Kaixin Li, Yuchen Tian, Qisheng Hu, Ziyang Luo, Zhiyong Huang, and Jing Ma. 2024b. Mmcode: Benchmarking multimodal large language models for code generation with visually rich programming problems. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 736–783.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. 2024a. Llava-next: Improved reasoning, ocr, and world knowledge.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2024b. Visual instruction tuning. *Advances in neural information processing systems*, 36.

Zihan Ma, Minnan Luo, Hao Guo, Zhi Zeng, Yiran Hao, and Xiang Zhao. 2024. Event-radar: Event-driven multi-view learning for multimodal fake news detection. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5809–5821.

Seungwhan Moon, Satwik Kottur, Paul Crook, Ankita De, Shivani Poddar, Theodore Levin, David Whitney, Daniel Difranco, Ahmad Beirami, Eunjoon Cho, Rajen Subba, and Alborz Geramifard. 2020. Situated and interactive multimodal conversations. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1103–1121, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Yao Mu, Junting Chen, Qing-Long Zhang, Shoufa Chen, Qiaojun Yu, GE Chongjian, Runjian Chen, Zhixuan Liang, Mengkang Hu, Chaofan Tao, et al. Robocodex: Multimodal code generation for robotic behavior synthesis. In *Forty-first International Conference on Machine Learning*.

Marius V Peelen and Sabine Kastner. 2011. A neural basis for real-world visual search in human occipitotemporal cortex. *Proceedings of the National Academy of Sciences*, 108(29):12125–12130.

Adina L Roskies. 1999. The binding problem. *Neuron*, 24(1):7–9.

Melanie Sclar, Gastón Bujia, Sebastián Vita, Guillermo Solovey, and Juan Esteban Kamienkowski. 2020. Modeling human visual search: A combined bayesian searcher and saliency map approach for eye movement guidance in natural scenes. *arXiv preprint arXiv:2009.08373*.

Shezheng Song, Shan Zhao, Chengyu Wang, Tianwei Yan, Shasha Li, Xiaoguang Mao, and Meng Wang. 2024. A dual-way enhanced framework from text matching point of view for multimodal entity linking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19008–19016.

Chen Su, Xinli Hu, Qingyan Meng, Linlin Zhang, Wenxu Shi, and Maofan Zhao. 2024. A multimodal fusion framework for urban scene understanding and functional identification using geospatial data. *International Journal of Applied Earth Observation and Geoinformation*, 127:103696.

Haozhe Wang, Chao Qu, Zuming Huang, Wei Chu, Fangzhen Lin, and Wenhu Chen. 2025. Vl-rethinker: Incentivizing self-reflection of vision-language models with reinforcement learning. *arXiv preprint arXiv:2504.08837*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Jeremy M Wolfe. 2020. Visual search: How do we find what we are looking for? *Annual review of vision science*, 6(1):539–562.

Jeremy M Wolfe, Melissa L-H Võ, Karla K Evans, and Michelle R Greene. 2011. Visual search in scenes involves selective and nonselective pathways. *Trends in cognitive sciences*, 15(2):77–84.

Penghao Wu and Saining Xie. 2024. V?: Guided visual search as a core mechanism in multimodal llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13084–13094.

Zhibo Yang, Lihan Huang, Yupei Chen, Zijun Wei, Seoyoung Ahn, Gregory Zelinsky, Dimitris Samaras, and Minh Hoai. 2020. Predicting goal-directed human attention using inverse reinforcement learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 193–202.

Barry Yao, Sijia Wang, Yu Chen, Qifan Wang, Minqian Liu, Zhiyang Xu, Licheng Yu, and Lifu Huang. 2024. Ameli: Enhancing multimodal entity linking with fine-grained attributes. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2816–2834, St. Julian's, Malta. Association for Computational Linguistics.

Mengmi Zhang, Jiashi Feng, Keng Teck Ma, Joo Hwee Lim, Qi Zhao, and Gabriel Kreiman. 2018. Finding any waldo with zero-shot invariant and efficient visual search. *Nature communications*, 9(1):3730.

A Benchmark Attributes

Table 6 contains the attribute list used in our benchmark, alongside the range of values each attribute can have.

B MATE Benchmark Dataset Format

Each instance of MATE contains key metadata for the definition of the scene, input prompt and key attributes needed to solve the task at hand. Table 7 describes these metadata.

C Participants in Human Evaluation

The participants in the human evaluation ranged from 28 to 44 years old and all had university degrees.

D All Evaluated Model Results

We evaluated a total of 16 models in our benchmark, MATE, from several families of VLMs. Table 9 summarizes the results we obtained with them, both in unimodal and crossmodal settings. It is an extended version of Tables 1 and 3.

E Predicted Object Attribute Overlapping in 3D Coordinate-Only Linking Attribute Cases

This section analyzes cases where 3D coordinates are the only linking attribute that uniquely identifies the target object. In Figure 5, we report how many attributes of Qwen2.5-VL's predicted objects match the target's.

Nº Attr.	Shape	Material	Color	Size	Acc.
3D Coords					48.7
1	✓				92.1
		✓			90.1
			✓		88.8
				✓	76.9
2	✓	✓			70.7
	✓			✓	54.8
		✓	✓		68.5
		✓		✓	58.0
			✓	✓	52.2
3	✓	✓		✓	44.2
		✓	✓	✓	57.1

Table 5: Average performance in instances where different linking attribute types are necessary to align objects across modalities, ordered by the number of attributes.

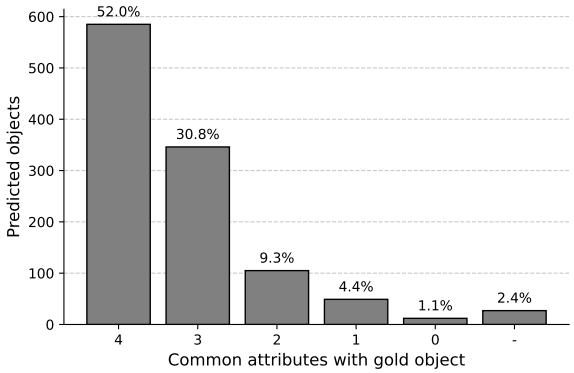


Figure 5: Number of matching attributes between Qwen2.5-VL's predicted object and the target object in cases where 3D coordinates are the only uniquely identifying linking attribute. A value of '4' indicates that the model selected the correct object (i.e., all attributes matched), while '–' indicates predictions that could not be linked to any existing object.

F Zero- and Few-shot Experiments

In Table 9 we show the results of all the VLMs for zero-, one- and two-shot prompting. Image-to-text and text-to-image exact match accuracies are provided for every variant.

G Linking Attribute Performance

Linking attributes have to be used to identify the pointed object in the target modality. Sometimes, one attribute is enough for this process, but other times combinations of two or three attributes are required. In Table 5, we show the average accuracy obtained for different combinations of attributes. Notice that 3D coordinates are treated apart since 3D coordinates are always unique.

H Evaluation prompts

In this appendix, we provide the specific prompts used to evaluate VLMs in this work.

H.1 UniModal Evaluation Prompts

Unimodal experiments are divided between text-to-text and image-to-image tasks. Figure 7 shows the prompt we use for text-to-text tasks, whereas Figure 6 shows the prompt for image-to-image tasks.

H.2 Cross-modal Evaluation Prompts

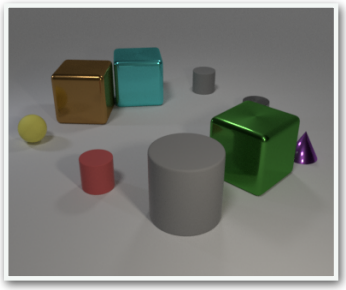
For cross-modal evaluation we also use two prompts, depending on the task variant (image-to-text or text-to-image). Figure 8 is used for image-to-text and Figure 9 for text-to-image.

Attr. name	Type	Values
Material	Discrete	{metal, rubber}
Shape	Discrete	{sphere, cube, cylinder, cone}
Color	Discrete	{blue, green, red, gray, cyan, brown, yellow, purple}
Name	Discrete	{Object_i} where $i \in [0, 9]$
Size	Discrete	{0.35, 0.351, 0.7, 0.701}
Rotation	Continuous	[0, 360]
3D coordinates	Continuous	$[x, y, z]$ where $(x, y, z) \in (-\infty, +\infty)$

Table 6: The complete list of attributes we use in the MATE dataset.

Name	Type	Description
example_id	str	Uniquely identifiable id of this instance.
task	str	Task of this example: img2img, txt2txt, img2txt or txt2img.
input_str	str	Input prompt that includes task definition, few-shot examples (when applicable) and question to be answered.
gold_reference	str	The expected answer by the model.
image	str	Filename of the scene's image.
scene	JSON	Serialized scene containing all objects and their attributes.
scene_format	str	Format used to represent the scene: JSON (by default), YAML, XML or TXT.
object_count	int	Number of objects in the scene (from 3 to 10).
pointer_attribute	JSON	Pointer attribute and its value.
target_attribute	JSON	Target attribute and its value.
key_attributes	list	Combination of attributes that uniquely identify the target object (not including the pointer and target attributes).
few_shot_attributes	list	Pointer, key and target attributes of few-shot examples.

Table 7: The complete list of metadata we use in the MATE dataset with detailed descriptions and data types.

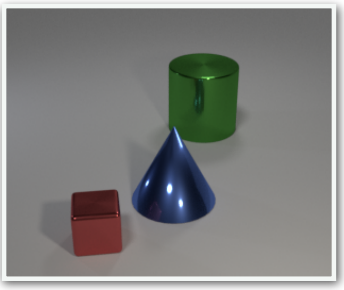


```
================================ Prompt ================================
This image shows a minimalist arrangement of 3D geometric shapes made
of different materials and colors.
Your job is to identify the cone in the image and determine its color
in the following format: {"answer": "COLOR"}
Select your answer from the following options: gray, red, blue, green,
brown, purple, cyan, or yellow. Do not add any additional text to your
answer, provide your answer always in the following format: {"answer":
"COLOR"}.

For example:
What is the shape of the cyan colored object? {"answer": "cube"}
What is the shape of the red colored object? {"answer": "cylinder"}
Now answer the following question:
What is the color of the cone?
```

Figure 6: Prompt used for image-to-image.

```
========================================== Prompt ==========================================
The following JSON contains information about all objects in a scene, including their name, 3D
coordinates (represented as [X, Y, Z]), material, and other attributes.
Your job is to analyze this information, find the object at coordinates [1.003104279700655,
-2.241594855367711, 0.699999988079071], and retrieve its name from its attributes in the following
format: {"answer": "NAME"}
Do not add any additional text to your answer, provide your answer always in the following format:
{"answer": "NAME"}.

Here is the JSON containing details about all objects in the scene in the image:

{"camera_location": [9.915478112284436, 0, 5.34366512298584], "objects": [{"name": "Object_0",
"shape": "cone", "size": 0.351, "material": "metal", "3d_coords": [0.5808469064750094,
2.3165597762151053, 0.3499999940395355], "rotation": 274.5267333984375, "color": "blue"}, {"name":
"Object_1", "shape": "cylinder", "size": 0.7, "material": "rubber", "3d_coords":
[1.003104279700655, -2.241594855367711, 0.699999988079071], "rotation": 108.92057037353516,
"color": "gray"}, {"name": "Object_2", "shape": "cube", "size": 0.351, "material": "rubber",
"3d_coords": [2.0921108024881967, 1.4308784617625303, 0.3499999940395355], "rotation":
9.969311714172363, "color": "yellow"}]}

For example:
What is the name of the object with a size of 0.7? {"answer": "Object_1"}
What is the name of the object with a rotation of 9.969311714172363? {"answer": "Object_2"}
Now answer the following question:
What is the name of the object at the coordinates [1.003104279700655, -2.241594855367711,
0.699999988079071]?
```

Figure 7: Prompt used for text-to-text.



```
============================== Prompt ==============================
This image shows a minimalist arrangement of 3D geometric shapes made
of different materials and colors. The JSON provided contains
information about all objects in the scene, including their name, 3D
coordinates (represented as [X, Y, Z]), material (metal = shinny,
rubber = matte), and other attributes.
Your job is to identify the green colored object in the image, match
its attributes to identify which of the objects mentioned in the JSON
corresponds to that object, and determine its name in the following
format: {"answer": "NAME"}
Do not add any additional text to your answer, provide your answer
always in the following format: {"answer": "NAME"}.

Understanding the Coordinate System X, Y, Z:
• X (Depth): Represents the depth relative to the camera. Smaller
values indicate objects that are farther away.
• Y (Horizontal Position): Represents the left-to-right position. A value of zero means the object
is centered in the scene, negative values place the object to the left, and positive values to the
right.
• Z (Vertical Position): Represents the height of the object's center point. Larger values
correspond to higher vertical positions.

Here is the JSON containing details about all objects in the scene in the image:

{"camera_location": [9.915478112284436, 0, 5.34366512298584], "objects": [{"name": "Object_0",
"shape": "cube", "size": 0.351, "material": "metal", "3d_coords": [3.2385542600785078,
-1.0702438583934402, 0.3499999940395355], "rotation": 110.09257507324219}, {"name": "Object_1",
"shape": "cone", "size": 0.7, "material": "metal", "3d_coords": [2.1338473656805106,
0.07773421879463616, 0.699999988079071], "rotation": 128.6031494140625}, {"name": "Object_2",
"shape": "cylinder", "size": 0.7, "material": "metal", "3d_coords": [-0.5097009561941948,
0.6912411302714815, 0.699999988079071], "rotation": 282.07916259765625}]}

For example:
What is the name of the red colored object? {"answer": "Object_0"}
What is the name of the cone? {"answer": "Object_1"}
Now answer the following question:
What is the name of the green colored object?
```

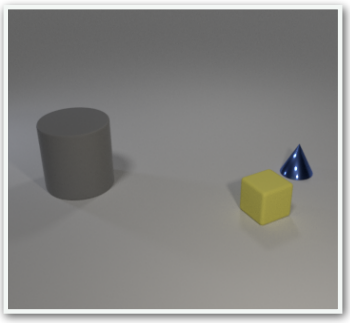
Figure 8: Prompt used for image-to-text.

Model	Unimodal			Crossmodal		
	Img2Img	Txt2Txt	Avg.	Img2Txt	Txt2Img	Avg.
Human	100.0	99.0	99.5	97.9	97.9	97.9
Random	18.7	25.3	22.0	25.4	18.5	22.0
llava-1.5-7b	45.7	60.7	53.2	18.2	27.5	22.9
llava-1.5-13b	55.3	88.6	71.9	29.3	35.7	32.5
llava-v1.6-mistral-7b	77.3	92.4	84.8	40.6	37.1	38.9
llava-v1.6-vicuna-7b	58.6	75.9	67.2	29.0	32.2	30.6
llava-v1.6-vicuna-13b	73.9	92.5	83.2	35.0	38.4	36.7
llava-v1.6-yi-34b	84.4	98.0	91.2	48.7	61.6	55.2
llama3-llava-next-8b	80.4	98.2	89.3	44.7	50.5	47.6
MolmoE-1B-0924	10.5	19.0	14.8	17.8	21.8	19.8
Molmo-7B-O-0924	88.3	71.9	80.1	32.3	31.4	31.8
Molmo-7B-D-0924	86.9	54.7	70.8	18.1	20.9	19.5
Llama-3.2-11B-Vision	68.9	97.1	83.0	37.4	11.4	24.4
Qwen2-VL-2B-Instruct	99.1	81.4	90.2	44.1	44.8	44.5
Qwen2-VL-7B-Instruct	**99.7**	98.1	98.9	72.1	77.2	74.7
Qwen2.5-VL-7B-Instruct	**99.7**	**99.4**	**99.5**	**75.7**	**84.5**	**80.1**
gemini-1.5-flash	95.9	**100.0**	**95.9**	63.2	71.2	67.2
gpt-4o-2024-11-20	**98.4**	**100.0**	**99.2**	76.4	79.1	77.8
claude-3-5-sonnet-20241022	97.3	**100.0**	98.7	**80.9**	**85.7**	**83.3**

Table 8: Results of open and closed VLMs in our task. All results are obtained using 2-shot prompting. Exact match accuracy is provided for image-to-image (Img2Img column), text-to-text (Txt2Txt column), image-to-text (Img2Txt column), and text-to-image (Txt2Img column) configurations. Human and random accuracies are shown as reference.

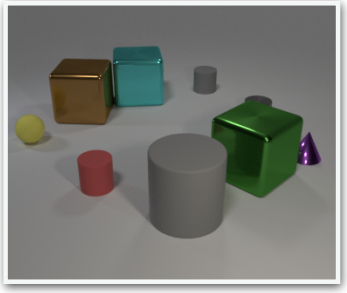
H.3 Chain-of-Thought

The prompts used for the chain-of-thought experiments are shown in Figure 10 (for the image-to-text task) and 11 (for text-to-image).



```
============================== Prompt ==============================
This image shows a minimalist arrangement of 3D geometric shapes
made of different materials and colors. The JSON provided contains
information about all objects in the scene, including their name, 3D
coordinates (represented as [X, Y, Z]), material (metal = shinny,
rubber = matte), and other attributes.
Your job is to analyze this information, find the object named
'Object_1', use its attributes to identify which object in the image
corresponds to that object, and retrieve its color in the following
format: {"answer": "COLOR"}
Select your answer from the following options: gray, red, blue,
green, brown, purple, cyan, or yellow. Do not add any additional
text to your answer, provide your answer always in the following
format: {"answer": "COLOR"}.

Understanding the Coordinate System X, Y, Z:
• X (Depth): Represents the depth relative to the camera. Smaller
values indicate objects that are farther away.
• Y (Horizontal Position): Represents the left-to-right position. A value of zero means the object
is centered in the scene, negative values place the object to the left, and positive values to the
right.
• Z (Vertical Position): Represents the height of the object's center point. Larger values
correspond to higher vertical positions.

Here is the JSON containing details about all objects in the scene in the image:

{"camera_location": [9.915478112284436, 0, 5.34366512298584], "objects": [{"name": "Object_0",
"shape": "cone", "size": 0.351, "material": "metal", "3d_coords": [0.5808469064750094,
2.3165597762151053, 0.3499999940395355], "rotation": 274.5267333984375}, {"name": "Object_1",
"shape": "cylinder", "size": 0.7, "material": "rubber", "3d_coords": [1.003104279700655,
-2.241594855367711, 0.699999988079071], "rotation": 108.92057037353516}, {"name": "Object_2",
"shape": "cube", "size": 0.351, "material": "rubber", "3d_coords": [2.0921108024881967,
1.4308784617625303, 0.3499999940395355], "rotation": 9.969311714172363}]}

For example:
What is the color of the object with a rotation of 274.52673339843755? {"answer": "blue"}
What is the color of the object with a rotation of 9.969311714172363? {"answer": "yellow"}
Now answer the following question:
What is the color of the object named 'Object_1'?
```

Figure 9: Prompt used for text-to-image.

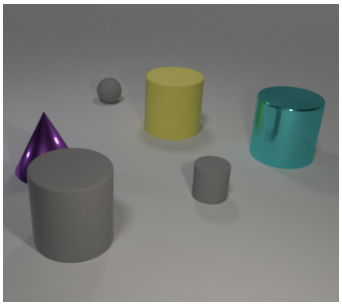


```
============================== prompt ==============================
This image shows a minimalist arrangement of 3D geometric shapes
made of different materials and colors. The JSON provided contains
information about all objects in the scene, including their name, 3D
coordinates (represented as [X, Y, Z]), material (metal = shinny,
rubber = matte), and other attributes.
Your job is to first identify the yellow colored object in the
image. Then, using only the attributes present in the objects in the
JSON, list the attributes of this object and list the ones that
distinguish this object from others in the image. Next, find the
object in the JSON that contains these same attributes. Finally,
return the value of the attribute 'name' of this object in the
following format: {"answer": "NAME"}

Understanding the Coordinate System X, Y, Z:
• X (Depth): Represents the depth relative to the camera. Smaller
values indicate objects that are farther away.
• Y (Horizontal Position): Represents the left-to-right position. A value of zero means the object
is centered in the scene, negative values place the object to the left, and positive values to the
right.
• Z (Vertical Position): Represents the height of the object's center point. Larger values
correspond to higher vertical positions.

Here is the JSON containing details about all objects in the scene in the image:

{JSON scene as in image-to-text}

For example:
What is the name of the red colored object? The red colored object in the image is a cylinder, made
of rubber, with an approximate size of 0.351, located around the coordinates [1.9566728749718734,
-1.5432421720941742, 0.3499999940395355] and is the only object in the image that is located around
the coordinates [1.9566728749718734, -1.5432421720941742, 0.3499999940395355]. In the JSON, the
name of the only object that matches the attributes {"3d_coords": [1.9566728749718734,
-1.5432421720941742, 0.3499999940395355]} and also contains the attributes {"shape": "cylinder",
"material": "rubber", "size": 0.351} is Object_1, therefore: {"answer": "Object_1"}
What is the name of the cyan colored object? {similar example}
Now answer the following question:
What is the name of the yellow colored object?
```

Figure 10: Prompt used for image-to-text with Chain-of-Thought.

Model	Image2Text			Text2Image		
	0	1	2	0	1	2
llava-1.5-7b	19.4	16.3	18.2	7.6	27.6	27.5
llava-1.5-13b	20.4	27.2	29.3	17.1	33.4	35.7
llava-v1.6-mistral-7b	36.7	38.1	40.6	34.6	36.3	37.1
llava-v1.6-vicuna-7b	26.3	28.0	29.0	11.4	29.6	32.2
llava-v1.6-vicuna-13b	26.1	32.7	35.0	7.7	33.4	38.4
llava-v1.6-yi-34b	43.2	46.8	48.7	54.5	55.9	61.6
Llama3-llava-next-8b	40.6	43.5	44.7	40.5	49.1	50.5
MolmoE-1B-0924	3.3	19.8	17.8	5.5	17.5	21.8
Molmo-7B-O-0924	31.4	24.9	32.3	19.1	33.8	31.4
Molmo-7B-D-0924	12.2	23.5	18.1	18.1	23.9	20.9
Llama-3.2-11B-Vision	27.1	36.1	37.4	4.0	12.1	11.4
Qwen2-VL-2B-Instruct	26.9	39.4	44.1	42.8	34.1	44.8
Qwen2-VL-7B-Instruct	68.8	72.4	72.1	77.4	77.9	77.2
Qwen2.5-VL-7B-Instruct	**72.0**	**74.8**	**75.7**	**83.6**	**82.9**	**84.5**
gemini-1.5-flash	59.1	61.9	63.2	63.6	68.9	71.2
gpt-4o-2024-11-20	74.8	75.5	76.4	78.8	80.2	79.1
claude-3-5-sonnet-20241022	**77.8**	**79.9**	**80.9**	**83.5**	**84.9**	**85.7**

Table 9: Results of open and closed VLMs in our task with zero, 1, and 2 shot prompting. Exact match accuracy is provided for image-to-text (Image2Text column) and text-to-image (Text2Image column) configurations.



```
============================== prompt ==============================
This image shows a minimalist arrangement of 3D geometric shapes
made of different materials and colors. The JSON provided contains
information about all objects in the scene, including their name, 3D
coordinates (represented as [X, Y, Z]), material (metal = shinny,
rubber = matte), and other attributes.
Your job is to first identify the object named 'Object_5' in the
JSON. Then, list the attributes of this object and list the ones
that distinguish this object from others in the JSON. Next, find the
object in the image that matches these same attributes. Finally,
return the color of this object in the following format: {"answer":
"COLOR"}
Select your answer from the following options: gray, red, blue,
green, brown, purple, cyan, or yellow.
Understanding the Coordinate System X, Y, Z:
• X (Depth): Represents the depth relative to the camera. Smaller
values indicate objects that are farther away.
• Y (Horizontal Position): Represents the left-to-right position. A
value of zero means the object is centered in the scene, negative values place the object to the
left, and positive values to the right.
• Z (Vertical Position): Represents the height of the object's center point. Larger values
correspond to higher vertical positions.

Here is the JSON containing details about all objects in the scene in the image:

{JSON scene as in image-to-text}

For example:
What is the color of the object named 'Object_1'? The object named 'Object_1' in the JSON also
contains the attributes {"shape": "cone", "material": "metal", "size": 0.7, "3d_coords":
[0.7603597884957096, -1.4732485834375681, 0.699999988079071]} and is the only object in the JSON
with the attributes {"shape": "cone"}. In the image, the color of the only object that is a cone
and is also made of metal, with an approximate size of 0.7, located around the coordinates
[0.7603597884957096, -1.4732485834375681, 0.699999988079071] is purple, therefore: {"answer":
"purple"}
What is the color of the object at the coordinates [1.752554189978086, 1.8608381882916412,
0.3499999940395355]? {similar example}
Now answer the following question:
What is the color of the object named 'Object_5'?
```

Figure 11: Prompt used for text-to-image with Chain-of-Thought.