Exploring Spatial Language Grounding Through Referring Expressions

Akshar Tumu[1,*], Parisa Kordjamshidi[2]

[1]University of California San Diego, 9500 Gilman Dr, La Jolla, CA, 92093, USA
[2]Michigan State University, East Lansing, MI, USA

Abstract

Spatial Reasoning is an important component of human cognition and is an area in which the latest Vision-language models (VLMs) show signs of difficulty. The current analysis works use image captioning tasks and visual question answering. In this work, we propose using the Referring Expression Comprehension task instead as a platform for the evaluation of spatial reasoning by VLMs. This platform provides the opportunity for a deeper analysis of spatial comprehension and grounding abilities when there is 1) ambiguity in object detection, 2) complex spatial expressions with a longer sentence structure and multiple spatial relations, and 3) expressions with negation ('not'). In our analysis, we use task-specific architectures as well as large VLMs and highlight their strengths and weaknesses in dealing with these specific situations. While all these models face challenges with the task at hand, the relative behaviors depend on the underlying models and the specific categories of spatial semantics (topological, directional, proximal, etc.). Our results highlight these challenges and behaviors and provide insight into research gaps and future directions.

Keywords

Spatial Reasoning, Vision-language models (VLMs), Referring Expression Comprehension

1. Introduction

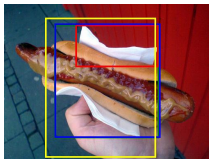
Vision-language model (VLM) research has boomed in the recent past, owing to the enhanced user interaction and accessibility they provide. Models such as GPT 4o[1], LLaVA [1], Google Gemini [2] have become adept at solving vision-language tasks such as Visual Question Answering (VQA), Image Captioning, and more. However, VLMs still lack human-level 'Spatial Reasoning' capabilities [3, 4, 5]. Spatial reasoning involves comprehending relations that depict the absolute/relative position or orientation of an object, such as 'left', 'above', or 'near'. Inaccurate spatial reasoning by VLMs can lead to serious consequences in embodied AI domains such as autonomous driving and surgical robotics. A focused analysis of VLMs' spatial reasoning capabilities can help identify and address potential reasoning issues.

Most of the previous works confine their analysis to testing which models work well for spatial relations. We go further to analyze the comparative performance of these models for spatial categories that represent different orientational and positional relations between objects. A novel aspect of our work is the analysis of the effect of varying spatial composition (number of spatial relations) in the expressions on the performance of the models.

Previous works focused on spatial analysis with image captioning-related tasks, thus failing to locate the source of error in the presence of visual and linguistic ambiguity. To avoid this, we adopt the Referring Expression Comprehension (REC) task for our analysis. The REC models output bounding boxes around the target entity based on a natural language expression, the analysis of which could reveal the parts of the input that the models fail to comprehend. Comprehension accuracy (or simply,

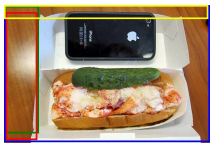
STRL'25: Fourth International Workshop on Spatio-Temporal Reasoning and Learning, 16 August 2025, Montreal, Canada
*Corresponding author.
atumu@ucsd.edu (A. Tumu); kordjams@msu.edu (P. Kordjamshidi)
https://www.cse.msu.edu/~kordjams/ (P. Kordjamshidi)
0009-0003-4883-8734 (A. Tumu)

[1]https://openai.com/index/hello-gpt-4o/



(a) The white napkin that is wrapped around the hot dog

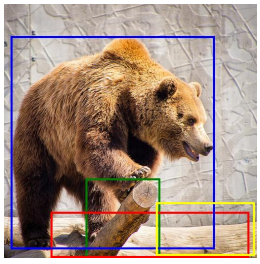
(b) The white box that is around the mirror



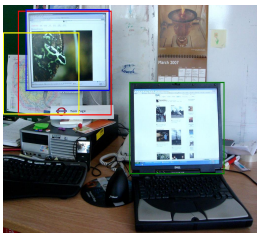
(c) The brown table that is to the left of the black cell phone

(d) The sandy shore that is near the murky water

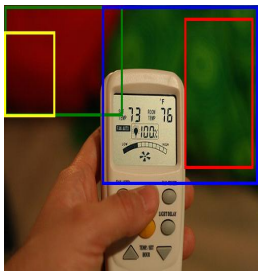
(e) The baseball player that is to the left of the black helmet and to the right of the home plate



(f) The large branch that is to the right of the log that is behind the large bear



(g) The black monitor that is to the left of the keyboard or on the desk



(h) The blanket that is not green and that is not on the bed

(i) The fence that is not black and that is not to the left of the man

Figure 1: Figures for Qualitative analysis. In the figures, the green box is the ground-truth bounding box. The red, blue, and yellow boxes are the output bounding boxes of MGA-Net, Grounding DINO, and LLaVA, respectively.

accuracy) is a common metric for this task; it captures how often a model correctly outputs the bounding box around the target entity.

For our analysis, we use the CopsRef dataset [6], which is a complex dataset with visual ambiguity and multiple spatial relations in expressions. We focus our analysis on 51 spatial relations, categorized into 8 categories.

We test two popular VLMs - LLaVA [1] and Grounding DINO [7]. We also include 'MGA-Net' [8], a model specifically designed for the REC task. The chosen models offer diversity in the evaluation as they differ in their architectural elements, training strategies, and input formats. We further compare these models with an object detector baseline to test if the images are truly complex and require elaborate referring expressions to ground the correct object.

Some of our important findings are as follows:
(1) Referring expressions that include spatial relations, in addition to object attributes, result in higher accuracy on the REC task compared to expressions with only attributes. **(2)** Increasing the spatial complexity (no. of spatial relations) of an expression affects the performance of the VLMs, but models with explicit compositional learning components maintain the performance. **(3)** Expressions involving dynamic spatial relations yield low accuracy across all models, indicating the difficulty in modelling these relations. **(4)** The task-specific trained models achieve higher accuracy for expressions with geometric spatial relations (e.g., left of, right of) while the VLMs show relatively better accuracy for expressions having ambiguous relations such as proximity. **(5)** The models fail to recognize negated spatial relations in referring expressions in multiple instances, though the extent of this failure varies across models.

Table 1
Statistics of Popular Referring Expression Comprehension Datasets. For the last column, the relation types are taken from various resources explained in Section 2 and [9], in addition to the relations in Table 2.

Dataset	Object Categories	Average length of expression	Average no. of objects per image	No. of spatial relations
RefCOCO	80	3.6	10.6	59
RefCOCOg	80	8.4	8.2	72
CLEVR-Ref+	3	22.4	6.5	4
CopsRef	508	14.4	17.4	51

2. Related Work

Previous works have conducted a broad analysis on the ability of VLMs to perform multimodal perception and reasoning tasks, such as Spatial Reasoning, Multimodal conversation, etc. Many comprehensive real-world benchmarks have been introduced to test multiple VLM capabilities. [1, 10, 11, 12, 13, 14].

Some works [4, 15] focus solely on spatial analysis of VLMs. SpatialEval benchmark [16] goes a step further to analyze the role of each modality in spatial reasoning. However, these works do not analyze the factors that affect the spatial reasoning ability of the VLMs. Another class of works performs a category-wise analysis of spatial relations, either based on their spatial properties [3, 17] or their linguistic properties and complexity [18]. Differing from these works, [5] analyzes the effects of spatial biases in the datasets for the REC task performance. Spatial analysis has also been approached through the use of text-only questions to probe pre-trained LLMs or VLMs [19, 20, 21]. However, these methods do not evaluate spatial grounding within the visual modality, which is a crucial aspect of our work.

Other closely aligned work includes Embodied Spatial Analysis, which focuses on the effects of different perspectives and non-verbal cues on the spatial reasoning capabilities of VLMs [22, 23].

Task Complexity and Interpretability. The works mentioned previously use image-caption agreement as their evaluation task. Due to the inherent limitations of this task, these works simplified the expressions to have only 2 objects and 1 spatial relation. To improve the interpretability of model output, synthetic datasets have been used instead of real-world images [4, 18, 24]. However, it simplifies the problem due to bounded expressivity (limited number of objects, attributes, and spatial relations). In our case, REC models output bounding boxes around the target objects. Analyzing the position and characteristics of the output object helps identify the parts of the input that the models fail to process. This enables comparative analysis of expressions with 0, 1, or more spatial relations, a unique feature of our work. The REC task also enables us to test the models over images of different visual complexities (single or multiple instances of objects in an image).

3. Dataset

Table 1 shows the key characteristics of some popular REC datasets. We chose CopsRef over RefCOCO and RefCOCOg due to the longer expression length and higher number of objects per image. Although CLEVR-Ref+ also provides a complex dataset, it is a synthetic dataset with limited expressiveness. Unlike other datasets, CopsRef's expressions go beyond describing the simple, distinctive properties of the objects. CopsRef is also a highly spatial dataset, as 90% of expressions consist of spatial relations. Examples of such referring expressions and the corresponding images are given in Figure 1.

Table 2 shows the category-wise split of the 51 spatial relations we identified in the CopsRef test dataset. We utilize the categories introduced by [9] and modify a few categories as per our dataset characteristics. For explanations of each category, refer to Appendix A. Note that while there are no referring expressions with only one relation from the orientation category, these relations co-occur with relations from other categories in some expressions.

Table 2
Category-wise relation split and number of referring expressions in the CopsRef test set with 1 spatial relation in each category

Category	Number	Spatial Relations
Absolute	56	on the right, on the left, in the middle, in the center, from the right, from the left
Adjacency	14	attached, against, on the side, on the back, on the front, on the edge
Directional	29	falling off, along, through, across, down, up, hanging from, coming from, around
Orientation	0	facing
Projective	2361	on top of, beneath, beside, behind, to the left, to the right, under, above, in front of, over, below, underneath
Proximity	217	by, close to, near
Topological	1054	connected, contain, with, surrounding, surrounded by, inside, between, touching, out of, at, in, on
Unallocated	56	next to, enclosing

Table 3
Frequency of occurrence of relations

Category	No. of relations	No. of expressions
None	0	1202
One	1	3787
Two-chained	2	1324
Two-and	2	3890
Two-or	2	2203
Three	3	180

Table 3 shows the number of expressions having 0, 1, 2, and 3 spatial relations. For expressions with 2 spatial relations, we have introduced three categories. The first category, 'Two-chained', consists of expressions in which spatial clauses are chained one after the other. The second category, 'Two-and', consists of expressions with two spatial clauses such that the referred object satisfies both of them. Finally, the 'Two-or' category consists of expressions with two spatial clauses such that the object referred to by the expression satisfies at least one of them. Figures 1e–g illustrate examples of the three categories.

4. Approach

In our analysis, we seek to answer the following research questions:
RQ1. Which spatial relation categories result in low accuracy for REC models? **RQ2.** How do different model characteristics/architectures influence the REC task accuracy for certain spatial relation categories compared to the others? **RQ3.** Does the inclusion of spatial relations increase or decrease the accuracy of REC models? **RQ4.** How does the number of spatial relations in the expressions affect the accuracy across different types of models? **RQ5.** Do the REC models accurately recognize negated spatial relations in expressions?

To answer these questions, we explain our research methodology and the designed experiments in this section.

4.1. Models Description

We select three distinct models for our analysis such that they differ in key components like architecture, pre-training tasks, and input formats.

MGA-Net. [8] It is an REC task-specific model whose compositional learning architecture was designed to handle complex expressions. It decomposes a query using the soft attention mechanism and processes visual and linguistic information using dedicated modules to construct a relational graph among objects. Then, it uses a Gated Graph Neural Network to perform multi-step reasoning over the referring expression. We first implement the Faster-RCNN model [25] to procure object proposals. Then, we generate the vector representations for these object proposals using a pre-trained ResNet-101 model. Considering the available computing resources, we omit the fourth (topmost) layer of the ResNet101 model to obtain a Partial CNN backbone. Finally, we train the model for ten epochs. We limit our training to ten epochs due to computational constraints.

Grounding DINO. [7] It is an open-set object detector VLM with language support. It has a vision and a language backbone whose outputs are fused at multiple levels. Its contrastive loss for grounded pre-training makes it suitable for the REC task. We use the Swin-B vision backbone and the CLIP-text encoder for the language backbone. We filter all bounding box detections for an expression using their output labels to see which detections match the target entity. Then we select the detection with the highest confidence score.

LLaVA. [1] It is a general-purpose VLM that connects an open-set vision encoder from CLIP [26] with a language decoder. The model is trained end-to-end, which involves visual instruction tuning for aligning the vision and language modalities. We test LLaVA with a **Short prompt**: (USER: <image>\n Give the bounding box for: "Referring Expression"\nASSISTANT:) and a **Long prompt**: (USER: <image>\n Provide the bounding box coordinates for the object described by the referring expression: "Referring Expression"\n ASSISTANT:). Both prompts have a similar structure, but the second prompt is longer.

OWL-ViT [27] It is an object detector baseline that only takes the target object's label as the input instead of the entire referring expression. It is an open-set object detector, which is required because CopsRef expressions involve entities from the Visual Genome [28] Scene Graphs, which have entities absent in common datasets used to train famous closed-set detectors like YOLO [29]. It also has a simple architecture with a Vision transformer and CLIP for aligning images and labels in a zero-shot manner, making it an ideal baseline.

Model Differences. A key difference in the three main models can be seen in their input format. While Grounding DINO and LLaVA take the entire image as the input and perform bounding box regression to get object proposals, MGA-Net directly takes the externally detected bounding boxes as the input. Grounding DINO and LLaVA also have similarities in their architectures, as they both have vision and language backbones that are fed the entire image and text inputs. This is unlike MGA-Net, which has dedicated transformer architecture modules for visual, linguistic, and relative location components. However, Grounding DINO and MGA-Net show similarities in having grounded training/pre-training tasks, while LLaVA only has general multimodal pre-training.

In addition to these three models, we also experimented with InstructBLIP [30] and OpenFlamingo [31] models for the REC task. These models are general-purpose VLMs. While InstructBLIP works in the zero-shot mode, OpenFlamingo functions in the few-shot mode. Neither of the models could provide meaningful outputs for the task. The outputs of these two VLMs have been discussed in more detail in Appendix B.

4.2. Experimental Setting and Evaluation

We create the following dataset test splits for evaluation and answering the earlier mentioned research questions, RQ1-RQ5.

4.2.1. Fine-grained Spatial Relations Split

In the test dataset, we split the expressions with 1 spatial relation using the categories shown in Table 2. Using the categories from Table 3, we split the remaining expressions based on the number of spatial relations they contain. Then, we rank the models based on their accuracy for each category.

To compare the models' performances across the categories, we employ a statistical test known as the Kendall Tau Independence Test. It evaluates the degree of similarity between two sets of ranks given to the same set of objects. We calculate the Kendall rank coefficient (τ), which yields the correlation between two ranked lists. Given τ value, we calculate the z statistic, which follows standard normal distribution, as:

$$z = 3 * \tau * \sqrt{n(n-1)}/\sqrt{2(2n+5)}. \quad (1)$$

Using the 2-tailed p-test at 0.05 level of significance, we test the following: **Null hypothesis**: There is no correlation between the two ranked lists. **Alternative hypothesis**: There is a correlation between the two ranked lists.

4.2.2. Visual Complexity Split

To observe the effect of visual complexity on model performance, we split the test dataset into two parts. The first part has images that have multiple instances of one or more objects mentioned in the associated referring expressions. The second part has images with at most one instance of every object mentioned in the expression. We perform this splitting by first collecting the entities in each expression using spaCy[2] and then employing Grounding DINO to find the number of instances in the image for each of the collected entities.

4.2.3. Negation Analysis Split

In our analysis, we found that models have difficulties in grounding spatial expressions with negations. Therefore, we created a test split for a more accurate evaluation and a deeper analysis of negated spatial expressions. We collected expressions that include the keyword 'not' and divided them into two sets according to the number of occurring negations (1 or 2). Then, we collected those expressions for which all three models give an IoU of less than 0.5. For each expression, we perform a qualitative analysis to verify whether the errors are due to misinterpreting the negations or conflation of other errors. We limit our analysis to the results from the first run of the three models to facilitate the instance-wise analysis.

5. Results

Hardware. For Grounding DINO, LLaVA, and the OWL-ViT baseline, we use the T4 GPU provided by Google Colaboratory for inference.[3] For MGA-Net, we use the NVIDIA GeForce GTX 1650 GPU for training. We run each model three times (both training and testing for MGA-Net, and inference for the VLMs and the baseline) to ensure the statistical significance of our results.

Evaluation Metrics. We evaluate the models using the Intersection over Union (IoU) metric. Following previous works [32, 33], we consider the output as a correct comprehension if the IoU is greater than 0.5. We calculate the *comprehension accuracy* (referred to as accuracy) as the fraction of data points that have an IoU >0.5.

Table 4
Comprehension Accuracies

Model	Accuracy (%)
MGA-Net (Partial CNN)	62.92 ± 0.11
Grounding DINO	70.93 ± 0.01
LLaVA - Short Prompt	34.96 ± 0.03
MGA-Net (Full CNN)	61.22 ± 0.15
LLaVA - Long Prompt	33.79 ± 0.01
OWL-ViT	56.34 ± 0

5.1. Evaluation on Referring Expressions

From Table 4, we can observe that Grounding DINO and MGA-Net outperform the OWL-ViT baseline, with the former achieving the highest accuracy in grounding the referring expressions. However, we also tried training MGA-Net with the full ResNet-101 visual backbone (Full CNN) instead of the partial backbone (Partial CNN). We could only train this model for four epochs due to computational constraints. However, the model crossed 60% test accuracy in just four epochs and was monotonically increasing. This shows that MGA-Net could potentially provide a better performance using adequate computational resources. To avoid unfair comparisons due to the training discrepancies, we focus our results on the relative performances of each model across different spatial relation categories rather than comparing the absolute performances.

For LLaVA, we used the prompts explained in Section 4.1. The shorter prompt gave a slightly better accuracy than the longer prompt. Hence, we used the shorter prompt for further experiments. The accuracy of LLaVA is less than both the other models and the baseline. Possible reasons are the lack of both bounding box regression and visual grounding instructions during pre-training.

Since we trained/tested each model for three runs, we report the average accuracy of the three runs and the standard deviation in the table. Since we re-train MGA-Net for each of these runs, there is a noticeable difference in model predictions in each run, leading to a slightly high standard deviation. However, we test the VLMs and the baseline zero-shot, leading to zero or near-zero standard deviation in the accuracies. This also follows for the future result tables.

5.2. Evaluation on Fine-grained Relations

Table 5
Category-wise accuracy and ranking

Category	MGA-Net	Rank	Grounding DINO	Rank	LLaVA	Rank
Absolute	70.24 ± 2.22	1	82.14 ± 0	2	44.64 ± 0	4
Adjacency	52.38 ± 3.37	12	78.57 ± 0	4	50 ± 0	1
Directional	52.87 ± 3.25	11	65.52 ± 0	12	27.59 ± 0	12
Projective	64.07 ± 0.08	4	69.12 ± 0	8	36.19 ± 0.08	6
Proximity	62.83 ± 0.22	8	80.65 ± 0	3	46.84 ± 0.22	3
Topological	67.32 ± 0.49	2	83.02 ± 0	1	48.51 ± 0.09	2
Unallocated	63.09 ± 0.84	6	75 ± 0	5	35.71 ± 0	7
None	62.39 ± 0.58	9	73.88 ± 0	6	42.89 ± 0.17	5
Two-chained	63.21 ± 0.22	5	70.67 ± 0.03	7	30.82 ± 0	9
Two-and	62.98 ± 0.09	7	68.45 ± 0.01	10	31.57 ± 0.07	8
Two-or	59.39 ± 0.21	10	68.97 ± 0.01	9	30.26 ± 0.04	10
Three	65.18 ± 2.1	3	67.78 ± 0	11	30.19 ± 0.26	11

[2]https://spacy.io/
[3]https://colab.research.google.com/

Table 5 shows a few general trends in results. The top 3-4 categories that each model performs the best for are categories with a single spatial relation. Among those, all three models perform well for the Topological and Absolute categories.

To answer **RQ1**, we observed that all the models give a low accuracy for expressions having Directional relations. A possible reason is that the spatial configurations of the involved objects are dynamic as they vary from image to image for the same spatial relation. This makes it difficult for the models to learn common patterns for recognizing these relations, resulting in low accuracy.

5.3. Impact of Multiple Spatial Relations

Table 6
Kendall Tau Independence Test results for category-wise ranks. Coeff: Kendall Rank Coefficient, GDINO: Grounding DINO.

Model 1	Model 2	Coeff, Z-score	2-tailed test (Correlated)
MGA-Net	GDINO	0.18, 0.51	No
MGA-Net	LLaVA	0.09, 0.26	No
GDINO	LLaVA	0.73, 2.05	Yes

Table 6 shows the Kendall Tau Independence test results for the three pairs of VLMs. We can observe that while the category-wise ranks of the VLMs (Grounding DINO and LLaVA) are correlated, MGA-Net's ranks aren't correlated with them. This motivates us to study the possible reasons behind the difference in the category-wise performances of MGA-Net and the VLMs.

Among spatial categories of MGA-Net and VLMs, the major difference occurs with the Proximity and Projective categories. To answer **RQ2**, we can observe that the 'Proximity' category ranks third for both the VLMs but 8th for MGA-Net. On the other hand, 'Projective' has a higher rank for MGA-Net than both VLMs. We can see that MGA-Net prefers geometric spatial relations like left of, on top of, etc., as it takes the relative locations of bounding boxes as input, which helps represent such relations. On the other hand, the two VLMs have a better ranking than MGA-Net for ambiguous relation categories that do not specify a clear distance or geometric direction (e.g., by, close to). This is because the vision backbones of the VLMs utilize the entire image and help capture relations between a region in the image and its surrounding regions, unlike MGA-Net, which only receives the detected bounding boxes as input.

Table 7
Relation frequency results and ranking

No. of relations	MGA-Net	Grounding DINO	LLaVA	OWL-ViT Baseline
None	62.39 ± 0.59	73.88 ± 0	42.89 ± 0.17	67.8 ± 0
One	64.85 ± 0.13	73.94 ± 0	40.33 ± 0.01	60.5 ± 0
Two	61.96 ± 0.07	69 ± 0	31.05 ± 0.06	52.39 ± 0
Three	65.18 ± 2.1	67.78 ± 0	30.19 ± 0.26	55 ± 0

To study further differences between MGA-Net and the VLMs, we design Table 7, which shows the performance of the three models and the OWL-ViT baseline for expressions having different numbers of spatial relations. We observe that VLMs perform considerably better for expressions with 0/1 spatial relations compared to expressions with 2/3 spatial relations. This proves that VLMs find it comparatively difficult to ground multiple spatial relations. However, MGA-Net takes advantage of its compositional learning architecture to handle multi-step reasoning, resulting in a similar performance for all categories.

An interesting observation is that the performance of the baseline considerably drops for the 'Two' and 'Three' categories, even though the spatial relations aren't being passed as input to the baseline.

The reason might be that 41.4% of these images have multiple instances of objects, the impact of which is explained in the next section.

From Table 7, we can also compare the performance of the models for expressions with none and one spatial relation. We observe that LLaVA performs better for the former and MGA-Net for the latter. Grounding DINO gives a similar performance for both.

Now, to answer **RQ3**, we observe in Table 5 that among the seven categories of single spatial relations, MGA-Net and Grounding DINO perform better for five of those compared to expressions with no spatial relations. LLaVA also performs better for four such categories. Thus, we can conclude that in a setup involving visual and linguistic ambiguity (such as ours), spatial relations along with visual attributes often aid the models in grounding the expressions, compared to the attributes alone. This is also reinforced by the results of the baseline. From Table 7, we can observe that while the baseline gives the second-best performance for expressions with no spatial relations, it drops to the third place for expressions with one spatial relation, with a 7.3% reduction in performance. This is because the baseline doesn't have access to the spatial relations.

Finally, Table 7 helps us answer **RQ4** as it shows the effect of increasing spatial relations on the performance of MGA-Net versus the VLMs (as discussed before).

5.4. Impact of Visual Complexity

Table 8
Results for accuracy in different visual complexity settings.

Model	Accuracy Single(%)	Accuracy Multi(%)
MGA-Net	64.91 ± 0.15	59.61 ± 0.04
G-DINO	72.54 ± 0.01	68.94 ± 0.01
LLaVA	37.69 ± 0.01	30.43 ± 0.1
DeepSeek-VL2	32.53 ± 0	27.46 ± 0
OWL-ViT	59.71 ± 0	51.3 ± 0

Out of 12586 test data points, we found that in the images of 4730 data points, there are multiple instances of objects mentioned in the referring expressions. Table 8 shows the accuracies of the three models and the OWL-ViT baseline for images with a single instance ('Accuracy Single' column) and multiple instances ('Accuracy Multi' column). The models perform better for the single instance images by 5.4% on average compared to the multi-instance images. The 8.4% performance drop of the baseline for multi-instance images proves that the images are indeed complex and require more than just the label as the input for grounding the right object. However, the 7.3% performance drop of LLaVA, as compared to MGA-Net and Grounding DINO, shows that grounded pre-training also plays a crucial role in helping the models ground the right object instance in multi-instance images.

5.5. Impact of Negation

Table 9
Results for negations in expressions.

	Two Negations	One Negation
Total	73	36
Total failure	59	24
Grounding DINO	58	23
LLaVA	29	17
MGA-Net	35	20

We obtained 36 expressions with 1 'not' and 73 expressions with 2 'not's for which all models gave incorrect predictions. Table 9 shows the total number of expressions we obtained with 1 and 2 negations. The 'Total failure' row gives the number of instances for which models failed to recognize at least 1 negation. We can observe that Grounding DINO has the highest number of failure instances. LLaVA performs better than Grounding DINO, possibly due to the Vicuna [34] language backbone, as it has a better language understanding (including negations) compared to Grounding DINO's CLIP text encoder. MGA-Net outperforms Grounding DINO since its training involves expressions with negations, increasing its ability to comprehend negations during testing. Hence, to answer **RQ5**, we observe that while all REC models face issues with recognizing negations, certain model characteristics and training paradigms might reduce the failure cases when expressions contain negations.

Table 10
Negation Metrics: MGA-Net vs. LLaVA

Models	Negations	Precision (%)	Recall (%)
MGA-Net	1	53.60	70.8
MGA-Net	2	41.38	51
LLaVA	1	64.54	47.23
LLaVA	2	60.35	41

Another interesting observation was for the outputs of MGA-Net and LLaVA models when they are close to the target object. From Table 10, we can see that while LLaVA has a better precision in such cases, MGA-Net has a better recall.

6. Qualitative Analysis

Here, we provide a qualitative analysis of certain issues faced by the models in handling referring expressions.

6.1. Directional Relations

The expressions pertaining to Figures 1a and 1b consist of the same spatial relation ('around'). In the first figure, the wrapping of the napkin around the hot dog only makes the napkin partially visible. But in the second figure, the white box around the mirror is almost entirely visible. This shows how the interpretation of 'around' is highly dependent on the configuration of the involved objects. For the first image, LLaVA fails to precisely localize the object, while MGA-Net only returns a part of the napkin that is visible. In the second image, both models fail to localize the object.

6.2. Projective and Proximity Relations

Figure 1c shows an example of Projective relations ('to the left'). MGA-Net succeeds in returning the correct part of the table that is to the left of the phone. While Grounding DINO simply returns the entire table, LLaVA identifies the wrong part. This shows the ability of MGA-Net to comprehend projective relations better, particularly when the target object is not apparent. An example of Proximity relations is in Figure 1d, where LLaVA and Grounding DINO return the shore that is 'near' the murky water, but MGA-Net fails to do so.

6.3. Multiple Spatial Relations

For 'Two-and' category expressions, the models sometimes only satisfy one of the spatial clauses. This often happens if multiple objects of the same class are in the image. For example, in Figure 1e, the output baseball player is to the left of the black helmet but is not to the right of the home plate.

Similarly, for 'Two-chained' category expressions, the models sometimes do not consider the entire expression. For example, in Figure 1f, MGA-Net and LLaVA return the 'log that is behind the large bear', and Grounding DINO returns the bear itself. None of the models consider the 'large branch' part of the expression, which should have been the output.

Finally, for 'Two-or' category expressions, the model might pay attention to only one spatial clause. Consequently, it returns an object satisfying that clause but not the additional attributes mentioned in the expression. For example, in Figure 1g, the model returns the monitor, which is to the 'left of the keyboard', but it does not satisfy the color attribute.

6.4. Negation

Figures 1h and 1i show two cases where all models fail to recognize negation. In 1h, we can observe that while MGA-Net is wrong, LLaVA is close to the ground truth but partially covers the target object (high precision, low recall). In 1i, while LLaVA is wrong, MGA-Net is closest to the ground truth but covers an excess area (low precision, high recall).

7. Conclusion

Spatial reasoning is an integral aspect of cognitive reasoning and embodied AI tasks. However, recent studies have shown that state-of-the-art VLMs often fail to accurately comprehend spatial relations. To better understand the limitations of these models, we evaluate their spatial understanding using the referring expression comprehension task because it requires explicit grounding of complex linguistic expressions in the visual modality. We picked multiple models, including Vision-language models (LLaVA, Grounding DINO) as well as task-specific models (MGA-Net). We observed that the VLMs that are trained in the wild with visual and textual data perform worse in grounding. All models show low accuracy in grounding Directional relations. However, the VLMs do better in vague relations such as proximity, while the task-specific models are better in geometrically well-defined relations such as left and right. While using spatial relations increases the grounding accuracy, using multiple relations makes the reasoning more challenging for all models, with a higher impact on VLMs. However, unlike VLMs, MGA-Net maintains its performance for complex spatial expressions due to its compositional learning architecture. In the presence of visual complexity, the performance of all models drops, but LLaVA's performance is affected the most due to a lack of grounded pre-training. Finally, both VLMs and task-specific models have failure cases when grounding expressions that include negation. These findings shed light on the gaps for future work on Vision-language models.

8. Future Directions

Although increasing the number of parameters of VLMs can improve their performance for expressions with simple spatial relations, architectural changes are necessary if the VLMs are to maintain their performance even for expressions with novel complex compositions of spatial relations. We observed that MGA-Net maintains a consistent performance for expressions with varying spatial complexity better than the VLMs due to its soft attention module, which decomposes the expression into its semantic components for compositional reasoning. This highlights the decomposition of complex spatial expressions as a potential path forward to help VLMs generalize. Alternative strategies [35] could be using multi-modal transformer models [36], [37] and techniques such as weight sharing across transformer layers or 'Pushdown layers' with recursive language understanding [38]. Another promising direction is Neuro-symbolic processing [39, 40], which involves generating symbolic programs from expressions using LLMs and conducting explicit symbolic compositions before grounding into visual modality. We plan to explore integrating such techniques with VLMs to improve their spatial compositional reasoning capabilities.

Another issue to address is the VLMs' inability to comprehend negations. MGA-Net's improved performance over Grounding DINO due to the presence of negated expressions in the training data motivates us to explore the augmentation of training/instruction tuning data of VLMs with synthetically generated negated expressions. Additionally, we also plan to formulate contrastive learning objectives to penalize the model when it fails to comprehend negations.

Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

References

[1] H. Liu, C. Li, Q. Wu, Y. J. Lee, Visual Instruction Tuning, Advances in neural information processing systems 36 (2024).

[2] G. Team, R. Anil, S. Borgeaud, Y. Wu, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, et al., Gemini: A Family of Highly Capable Multimodal Models, arXiv preprint arXiv:2312.11805 (2023).

[3] F. Liu, G. Emerson, N. Collier, Visual Spatial Reasoning, Transactions of the Association for Computational Linguistics 11 (2023) 635–651.

[4] S. Subramanian, W. Merrill, T. Darrell, M. Gardner, S. Singh, A. Rohrbach, ReCLIP: A Strong Zero-Shot Baseline for Referring Expression Comprehension, arXiv preprint arXiv:2204.05991 (2022).

[5] A. Kamath, J. Hessel, K.-W. Chang, What's "up" with vision-language models? Investigating their struggle with spatial reasoning, arXiv preprint arXiv:2310.19785 (2023).

[6] Z. Chen, P. Wang, L. Ma, K.-Y. K. Wong, Q. Wu, Cops-Ref: A new Dataset and Task on Compositional Referring Expression Comprehension, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 10086–10095.

[7] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, C. Li, J. Yang, H. Su, J. Zhu, et al., Grounding DINO: Marrying DINO with Grounded Pre-Training for Open-Set Object Detection, arXiv preprint arXiv:2303.05499 (2023).

[8] Y. Zheng, Z. Wen, M. Tan, R. Zeng, Q. Chen, Y. Wang, Q. Wu, Modular Graph Attention Network for Complex Visual Relational Reasoning, in: Proceedings of the Asian Conference on Computer Vision, 2020.

[9] C. K. Marchi Fagundes, K. Stock, L. S. Delazari, A cross-linguistic study of spatial location descriptions in New Zealand English and Brazilian Portuguese natural language, Transactions in GIS 25 (2021) 3159–3187.

[10] Y. Tian, T. Ma, L. Xie, J. Qiu, X. Tang, Y. Zhang, J. Jiao, Q. Tian, Q. Ye, ChatterBox: Multi-round Multimodal Referring and Grounding, arXiv preprint arXiv:2401.13307 (2024).

[11] Y. Liu, H. Duan, Y. Zhang, B. Li, S. Zhang, W. Zhao, Y. Yuan, J. Wang, C. He, Z. Liu, et al., MMBench: Is Your Multi-modal Model an All-around Player?, arXiv preprint arXiv:2307.06281 (2023).

[12] B. Li, R. Wang, G. Wang, Y. Ge, Y. Ge, Y. Shan, SEED-Bench: Benchmarking Multimodal LLMs with Generative Comprehension, arXiv preprint arXiv:2307.16125 (2023).

[13] W. Yu, Z. Yang, L. Li, J. Wang, K. Lin, Z. Liu, X. Wang, L. Wang, MM-Vet: Evaluating Large Multimodal Models for Integrated Capabilities, arXiv preprint arXiv:2308.02490 (2023).

[14] X. Fu, Y. Hu, B. Li, Y. Feng, H. Wang, X. Lin, D. Roth, N. A. Smith, W.-C. Ma, R. Krishna, BLINK: Multimodal Large Language Models Can See but Not Perceive, in: European Conference on Computer Vision, Springer, 2025, pp. 148–166.

[15] P. J. Rösch, J. Libovickỳ, Probing the Role of Positional Information in Vision-Language Models, arXiv preprint arXiv:2305.10046 (2023).

[16] J. Wang, Y. Ming, Z. Shi, V. Vineet, X. Wang, Y. Li, N. Joshi, Is A Picture Worth A Thousand Words?

Delving Into Spatial Reasoning for Vision Language Models, arXiv preprint arXiv:2406.14852 (2024).
[17] T. Gokhale, H. Palangi, B. Nushi, V. Vineet, E. Horvitz, E. Kamar, C. Baral, Y. Yang, Benchmarking Spatial Relationships in Text-to-Image Generation, arXiv preprint arXiv:2212.10015 (2022).
[18] A. Kuhnle, H. Xie, A. Copestake, How clever is the FiLM model, and how clever can it be?, in: Proceedings of the European Conference on Computer Vision (ECCV) Workshops, 2018, pp. 0–0.
[19] A. G. Cohn, J. Hernandez-Orallo, Dialectical language model evaluation: An initial appraisal of the commonsense spatial reasoning abilities of LLMs, arXiv preprint arXiv:2304.11164 (2023).
[20] X. Liu, D. Yin, Y. Feng, D. Zhao, Things not Written in Text: Exploring Spatial Commonsense from Visual Signals, arXiv preprint arXiv:2203.08075 (2022).
[21] R. Mirzaee, H. R. Faghihi, Q. Ning, P. Kordjmashidi, SPARTQA: A Textual Question Answering Benchmark for Spatial Reasoning, arXiv preprint arXiv:2104.05832 (2021).
[22] M. M. Islam, R. Mirzaiee, A. Gladstone, H. Green, T. Iqbal, CAESAR: An Embodied Simulator for Generating Multimodal Referring Expression Datasets, Advances in Neural Information Processing Systems 35 (2022) 21001–21015.
[23] M. M. Islam, A. Gladstone, R. Islam, T. Iqbal, EQA-MX: Embodied Question Answering using Multimodal Expression, in: The Twelfth International Conference on Learning Representations, 2023.
[24] M. Lewis, N. V. Nayak, P. Yu, Q. Yu, J. Merullo, S. H. Bach, E. Pavlick, Does CLIP Bind Concepts? Probing Compositionality in Large Image Models, arXiv preprint arXiv:2212.10537 (2022).
[25] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, IEEE transactions on pattern analysis and machine intelligence 39 (2016) 1137–1149.
[26] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al., Learning Transferable Visual Models From Natural Language Supervision, in: International conference on machine learning, PMLR, 2021, pp. 8748–8763.
[27] M. Minderer, A. Gritsenko, A. Stone, M. Neumann, D. Weissenborn, A. Dosovitskiy, A. Mahendran, A. Arnab, M. Dehghani, Z. Shen, et al., Simple Open-Vocabulary Object Detection with Vision Transformers, in: European Conference on Computer Vision, Springer, 2022, pp. 728–755.
[28] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, et al., Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations, International journal of computer vision 123 (2017) 32–73.
[29] J. Redmon, You Only Look Once: Unified, Real-Time Object Detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016.
[30] W. Dai, J. Li, D. Li, A. Tiong, J. Zhao, W. Wang, B. Li, P. Fung, S. Hoi, InstructBLIP: Towards General-purpose Vision-Language Models with Instruction Tuning, arXiv preprint arXiv:2305.06500 2 (2023).
[31] A. Awadalla, I. Gao, J. Gardner, J. Hessel, Y. Hanafy, W. Zhu, K. Marathe, Y. Bitton, S. Gadre, S. Sagawa, et al., OpenFlamingo: An Open-Source Framework for Training Large Autoregressive Vision-Language Models, arXiv preprint arXiv:2308.01390 (2023).
[32] L. Yu, Z. Lin, X. Shen, J. Yang, X. Lu, M. Bansal, T. L. Berg, MAttNet: Modular Attention Network for Referring Expression Comprehension, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 1307–1315.
[33] J. Chen, F. Wei, J. Zhao, S. Song, B. Wu, Z. Peng, S.-H. G. Chan, H. Zhang, Revisiting referring expression comprehension evaluation in the era of large multimodal models, arXiv preprint arXiv:2406.16866 (2024).
[34] novita.ai, Vicuna: an Open-Source Large Language Model for Chatbots, https://blogs.novita.ai/vicuna-an-open-source-large-language-model-for-chatbots/, 2024. Published: 2024-04-18. Accessed: 2024-07-26.
[35] S. Sinha, T. Premsri, P. Kordjamshidi, A Survey on Compositional Learning of AI Models: Theoretical and Experimental Practices, arXiv preprint arXiv:2406.08787 (2024).
[36] A. Sikarwar, A. Patel, N. Goyal, When Can Transformers Ground and Compose: Insights from

Compositional Generalization Benchmarks, arXiv preprint arXiv:2210.12786 (2022).
[37] L. Qiu, H. Hu, B. Zhang, P. Shaw, F. Sha, Systematic Generalization on gSCAN: What is Nearly Solved and What is Next?, arXiv preprint arXiv:2109.12243 (2021).
[38] S. Murty, P. Sharma, J. Andreas, C. D. Manning, Pushdown Layers: Encoding Recursive Structure in Transformer Language Models, arXiv preprint arXiv:2310.19089 (2023).
[39] D. Kamali, E. J. Barezi, P. Kordjamshidi, NeSyCoCo: A Neuro-Symbolic Concept Composer for Compositional Generalization, arXiv preprint arXiv:2412.15588 (2024).
[40] J. Hsu, J. Mao, J. Tenenbaum, J. Wu, What's Left? Concept Grounding with Logic-Enhanced Foundation Models, Advances in Neural Information Processing Systems 36 (2024).

A. Description of spatial categories

For our analysis, we utilize the spatial categories introduced by [9] and replace the 'Cardinal Direction' category with 'Absolute'. The descriptions and examples for the chosen categories are as follows:

1. **Absolute**: Consists of relations that describe the location of an object in an absolute manner and not in relation to another object.
   E.g.: man on the right that is standing and wearing gray pant
2. **Adjacency**: Consists of relations that describe the close, side-by-side positioning of two objects. They may or may not imply a particular direction.
   E.g.: The large poster that is leaning against the wall
3. **Directional**: Consists of dynamic action verbs / directional relations. They describe the movement or change in position of an object relative to other objects in the image. The interpretation of these relations heavily relies on the configuration of the involved objects and/or the dynamic spatial relationship between them.
   E.g.: The gray car that is driving down the road
4. **Orientation**: Consists of relations which describe the orientation of an object w.r.t another object.
   E.g.: The sitting dog that is facing the window that is to the right of the mirror
5. **Projective**: Consists of relations that indicate the concrete spatial relationship between two objects, i.e., these relations can be quantified in terms of the coordinates of the two objects.
   E.g.: The black oven that is above the drawer
6. **Proximity**: Consists of relations that indicate that two objects are near each other without giving a specific directional relationship.
   E.g.: The blue chair that is close to the white monitor
7. **Topological**: Consists of relations that indicate the broader arrangement or the containment of an object w.r.t another object
   E.g.: The silver train that is at the colorful station
8. **Unallocated**: Consists of relations that cannot be allocated to any of the above categories.

B. Experiments with other VLMs

In our analysis, we also experimented with InstructBLIP [30] and OpenFlamingo [31] models for the REC task. These models are general-purpose VLMs with InstructBLIP working in the zero-shot model and OpenFlamingo in the few-shot mode. In this section, we discuss the prompts that we used for these two models and the outputs obtained for the prompts:

B.1. InstructBLIP

For InstructBLIP, we designed three prompts for the REC task. They are as follows:

1. Bounding Boxes: bounding box list; Referring Expression: Refexp; The index of the output bounding box is:
2. Bounding Boxes: bounding box list; Referring Expression: Refexp; The coordinates of the output bounding box are:
3. Provide the bounding box coordinates for: "Refexp"

In the prompts, the 'bounding box list' placeholder takes the coordinates of the detected bounding boxes in the image being passed as the input, along with indices for each bounding box, starting from '1'. But for the third prompt, the model has no access to pre-detected candidate bounding boxes in the image. While the expected output for the first prompt is the index of the correct bounding box, for the other 2 prompts it is the bounding box coordinates as the output.

The bounding box format is [x1, y1, x2, y2], where (x1, y1) is the bottom left corner and (x2, y2) is the top right corner of the box. The coordinate values are a fraction of the total length/width of the image according to the position of the coordinate.

Unfortunately, none of the prompts gave consistently correct outputs. The outputs were as follows:
Prompt 1: The outputs were mostly incorrect. Sometimes, the model also gave '0' as the output, even though it is not a valid index.
Prompt 2: The output did not return meaningful coordinates in most cases. But in the few instances that it did, they were mostly incorrect. Example outputs when the model could not return meaningful coordinates are:

- `{1: [0.16, 0.55], 2: [0.32, 0.47], 3: [0.55, 0.6], 4: [0.21, 0.06]}`
- `[0.9, 0.53, 0.93, 0.57, 0.0, 0.39]`

Prompt 3: The model could not understand the task, and it just paraphrased parts of the prompt instead of giving the coordinates as the output. Example prompts and outputs are:

- **Prompt:** Provide the bounding box coordinates for: "The large poster that is leaning against the wall"
  Output: what is the bounding box coordinates for the large poster that is leaning against the wall
- **Prompt:** Provide the bounding box coordinates for: "The young man that is leaning against the wall"
  Output: is standing in an elevator. the young man that is leaning against the wall is standing in an elevator

B.2. OpenFlamingo

We tested all the prompts designed for OpenFlamingo in both 2 and 3-shot settings.

Prompt 1:

- **Example output format:** <image>Bounding Boxes:bounding box list; Expression: Refexp; Correct Bounding Box:"ID"<|endofchunk|>
- **Query format:** <image>Bounding Boxes:bounding box list; Expression: Refexp; Correct Bounding Box:"

'bounding box list' placeholder takes the list of candidate bounding boxes in the image as input, in the same format as InstructBLIP (discussed in the previous section). The expected output is the index of the correct bounding box. However, we observed that irrespective of the query, the model gave the same output index for the same set of prompting examples.

Prompt 2:

- **Example output format:** <image>Expression: Refexp; Correct Bounding Box:[Bounding box coordinates]<|endofchunk|>

- **Query format:** <image>Expression: Refexp; Correct Bounding Box:[

'bounding box list' placeholder takes the same input as explained for Prompt 1. But instead of expecting the index, we expect the coordinates of the bounding box as the output. The format of the bounding box is the same as explained for InstructBLIP in the previous section. However, the model failed to give meaningful coordinates as output in most cases. When it did give meaningful coordinates, the outputs were mostly incorrect.