11777 Final - Understanding Compositionality in Vision-Language Models

Abhinav Rao [*1] **Akhila Yerukola** [*1] **Jean de Dieu Nyandwi** [*1] **Simran Khanuja** [*1]

Abstract

Understanding multimodal compositionality requires a comprehensive grasp of various elements including objects, their relations and attributes, order sensitiveness and overall deep atom-level understanding of both images and textual captions. Despite advancements in the unimodal performance of visual and text encoders, effectively modeling cross-modal interactions—in addition to unimodal entities and relationships remains a complex aspect of multimodal compositionality. In our project, we adopt a bottom-up approach, initially identifying important types of meta-data extractable from each modality, and then identifying challenges faced by current unimodal and multimodal models. To overcome the challenges faced by multimodals on image-text pairs with high unimodal similarity in WinoGround, we construct a synthetic dataset of hard-negatives for a more comprehensive examination of cross-modal interactions. Also, we leverage code generation and phrase grounding generation capabilities of LLMs in visual programming approaches to address the challenges of visio-linguistic compositionality. Furthermore, based on the performed analysis, we aim to enhance multimodal interactions via phase grounding. By generating synthetic hard-negative samples and finetuning both BLIP-2 and CLIP on generated data, we observe improvements in both models on text retrieval, image retrieval, and both(group score). We also observes a boost in performance in samples with high intra-caption, intra-image similarity, and samples with low AMR(Abstract Meaning Representation) length.

[*]Equal contribution [1]Carnegie Mellon University. Correspondence to: Abhinav Rao <abhinavr@cs.cmu.edu>, Akhila Yerukola <ayerukol@cs.cmu.edu>, Jean De Dieu Nyandwi <jeandedi@cs.cmu.edu>, Simran Khanuja <skhanuja@cs.cmu.edu>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

1. Introduction

Compositionality is the understanding that *the meaning of the whole is a function of the meanings of its parts* (Cresswell, 1973). In language, the whole is a sentence made of words; and in vision, the whole is a scene made up of objects, their attributes and relations (Ma et al., 2023). Despite it being a key characteristic of human intelligence, vision-language models struggle to perform well on compositional benchmarks, even with massive scale.

Compositionality is a long-tail problem in visual language models and it is the ultimate testbed of VLMs. How VLMs perform on compositional reasoning tasks is a good indication of their representation learning and generalization capabilities. There are several benchmarks that evaluate the compositional nature of foundation models across axes such as systematicity and productivity (Ma et al., 2023), multilingual and socio-cultural data (Liu et al., 2021), and word-ordering (Thrush et al., 2022). Additionally, studies have attempted to map the compositional reasoning capabilities of a model to its architectural components (Sikarwar et al., 2022). However, the problem of compositionality is challenging for both the textual modality (Berglund et al., 2023; Zhou et al., 2023), and for multimodal systems (Ma et al., 2023). A recent study (Diwan et al., 2022) has shown that the main reason why VLMs fail at compositional reasoning lie in fusion of image and text representations. Another plausible reason may come from the fact that current compositional datasets are very challenging and may not necessarily reflect the skills that VLMs learn during pre-training. Another recent work has shown that long captions hurt compositional performance and the model size and size of training data have no impact on compositional reasoning (Ma et al., 2023). While those are sounds reasons, they are not established yet and understanding compositionality is an ongoing challenge in VLMs.

In this project, we adopt a bottom-up approach, first identifying and extracting key types of meta-data information from each modality—images and textual captions. Subsequently, we compare and contrast the performance of unimodal and multimodal baselines. More specifically, for multimodal baselines, we draw comparisons against rich unimodal encoders featuring minimal multimodal interaction, shallow unimodal encoders with a high degree of multimodal interac-

tion, and large-scale frozen encoders undergoing learnable multimodal interactions. We analyse the performance of these baseline models on a number of dimensions (Section 6). This includes intra-caption similarity, intra-image similarity, AMR tree depth, Object-Relation count, and phrase grounding capabilities.

Finally, we use these analyses to guide our research towards the importance of explicitly enhancing multimodal interactions via learning grounding phrases. Next, we are interested in developing a synthetic test set with carefully generated hard-cases for images and matching texts. This will enable us to accurately assess the robustness of image similarity and text caption similarity in current models. Further, we are interested in leveraging the code generation and phrase grounding generation capabilities of large language models in visual programming approaches to combine the effectiveness of grounded phrases and compositional vision modules for effective visual-linguistic compositional tasks.

Code and Visualization: Our code can be found here: https://github.com/simran-khanuja/11777-project. We have uploaded all models, their performance, metadata and slices as described in Section 6 on a the zeno platform here: https://tinyurl.com/zeno-group.

2. Related Work

2.1. Neuro-symbolic Approaches for VLM Compositionality

Early work on Neuro Module Networks (Andreas et al., 2016; Johnson et al., 2017) argues that complex vision tasks can be divided into smaller perceptual units and are inherently compositional. Visual procedure programming, which is connected to neuro-symbolic methods (Hu et al., 2017; Yi et al., 2018), has explored the separation of reasoning from perception. However, these approaches require extensive supervision and end-to-end training, presenting unique training challenges and limited generalizability across domains.

Recently, neuro-symbolic approaches have regained popularity with the adoption of visual programming style approaches using large language models (LLMs). They are now being used commonly for vision-and-language tasks, allowing control over external visual modules and combining their outputs to generate a final response (Yang et al., 2022; Hu et al., 2022; Wu et al., 2023). Visual programs typically consist of a sequence of steps, each detailing which module will be utilized. These modules enable capabilities such as image understanding, image manipulation, and knowledge retrieval. An interpreter then handles the program execution, processing each line and continually updating the program state accordingly. This approach was reintroduced by VisProg (Gupta & Kembhavi, 2023) using LLMs, which generates pseudocode instructions and interprets them as a visual program, requiring only a small number of in-context examples. ViperGPT (Surís et al., 2023) takes a similar approach by directly generating unrestricted Python code, leveraging the code generation capabilities of LLMs. Moreover, visual programming has expanded its application to text-to-image generation (Cho et al., 2023) and compositional audio creation (Liu et al., 2023b), well utilizing the powerful capabilities of both LLMs and multimodal modules.

However, the usage of LLMs presents its own set of challenges. Since our LLMs were primarily trained with English-heavy datasets, they might not understand prompts written in non-English languages. Further, generating evaluation programs using LLMs can be costly both in terms of computational resources and price.

2.2. End-to-end Approaches for VLM Compositionality

VLMs learn joint image-text representations that generalize to a wide range of visual-language downstream tasks such as image captioning and visual question answering. These models are typically pre-trained on web-scale datasets (Changpinyo et al., 2021; Jia et al., 2021; Schuhmann et al., 2021), on unimodal objectives such as masked image or language modeling, and multi-modal objectives such as contrastive image-text retrieval (Radford et al., 2021) and image-text matching (Li et al., 2022).

VLMs are either dual-encoder based, or encoder-decoder based. Encoder-based models, with separate encoders for images and text (Radford et al., 2021; Jia et al., 2021), are good for image retrieval but lack reasoning due to limited fusion between modalities. Past works have addressed this via fusion networks that explicitly capture higher-order multimodal interactions (Chen et al., 2019; Li et al., 2021). Encoder-decoder models, pre-trained using generation objectives, capture stronger multimodal interactions, and hence reason well even in zero-shot and few-shot settings (Wang et al., 2021; Alayrac et al., 2022).

Despite recent trends of massively scaled VLMs(Wang et al., 2021; Alayrac et al., 2022; Yu et al., 2022; Chen et al., 2022) that achieve exceptionally high performance on multiple tasks, compositional reasoning remains to be a failure point. For instance, the top-3 best-performing models on Winoground are VQ2, PaLI, BLIP-2, but their accuracies are less than 50% [1].

3. Research Questions

Based on our midterm analysis, we identify four key factors that have a strong correlation with performance:

- *Observation 1:* All baseline models' performance is

[1] https://tinyurl.com/viswinoground

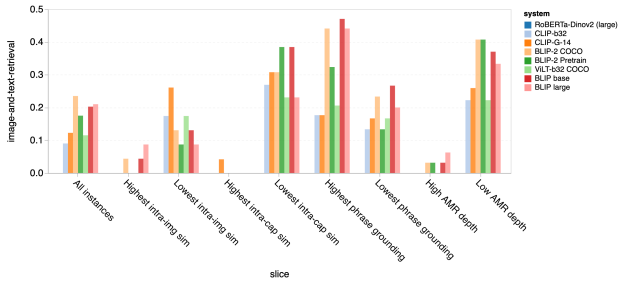

Figure 1. Overview of the performance of each slice as explained in §3

inversely correlated with intra-image similarity calculated using CLIP-ViT.

- *Observation 2:* All baseline models' performance is inversely correlated with intra-caption similarity calculated using RoBERTa.
- *Observation 3:* All baseline models' performance is directly correlated with phrase-grounding scores calculated using GroundingDINO.
- *Observation 4:* All baseline models' performance is inversely correlated with the degree of semantic compositionality in the caption (captured by calculating the depth of a semantic tree representation of the sentence).

All of these observations can be visualized in 3. Here we create slices of data from Winoground containing pairs with highest and lowest intra-image similarity (*Observation 1*), highest and lowest intra-caption similarity (*Observation 2*), highest and lowest phrase-grounding scores (*Observation 3*), and highest and lowest AMR depth (*Observation 4*). This analysis informs the research questions we ask below. We leverage these insights to make models robust to variation along all of these dimensions.

RQ1: Can we make models more robust to variations in intra-image and intra-caption similarity? How can we further separate maximally similar pairs in the embedding space?

RQ2: Would enhancing the phrase-grounding capabilities of models have a downstream effect on improved performance in visio-linguistic compositional reasoning?

RQ3: Can we break down highly compositional captions to do atomic-level reasoning in an iterative fashion, rather than reasoning about a long caption in one step? How can visual information further enhance multi-step reasoning?

We present our proposed approaches to tackle each of these research questions below.

4. Proposed Approaches

4.1. RQ1: Parallel Synthetic Data Creation for Robustness

According to the conducted error analysis, the performance of VLMs drops for samples with high intra-caption and high intra-image similarity. To improve the robustness of VLMs on those samples, we generate a synthetic dataset that contains captions and images that share a similarity to that of winoground (i.e. positive pair augmentation), and we fine-tune select models on the dataset. The generated 'harder' captions and images are semantically close to the original captions and images.

Augmenting existing captions (Hsieh et al., 2023; Yuksekgonul et al., 2023) and images (Li et al., 2020) based on the captions have been explored in previous works; but a combined approach targeting compositionality hasn't yet been explored. Furthermore our synthetic data generation keeps augmented caption and images semantically close to the original: we do not introduce new words into the captions, the generated image mimics the scene of the original image, and the resulting data generation pipeline is robust, simple, and scalable. Below, we expand on the steps involved in generating captions and images.

4.1.1. Generating Hard Captions

Caption augmentation is the very first step in the data generation pipeline. Given an existing caption from MS-COCO captions, we aim to generate multiple variants of the caption and then retain those that are closely similar to original captions and yet coherent and logical. Our idea is to first identify the adjectives and nouns in the caption and swap the adjectives with each other. We aim to perform the caption augmentation on sentences with two adjectives and nouns, which complement our goal of building a robust and effective synthetic dataset of hard cases.

To identify the adjectives and nouns, we use the SpaCy toolkit. Given the original caption C_o, the augmented caption C_{aug} is obtained by swapping the adjectives with each other, ensuring that the new caption is semantically aligned with the original caption and syntactically correct.

4.1.2. Generating Hard Images

As stated previously, VLMs struggle on pairs of images that have high-intra-image similarity and to combat that, we aim to generate 'harder' images that maintain original scenes but with different layouts of objects. To ensure the image generated aligns with the augmented caption, the augmented caption serves as the guiding instructions in image generation.

In the proposed approach, generating images is an iterative

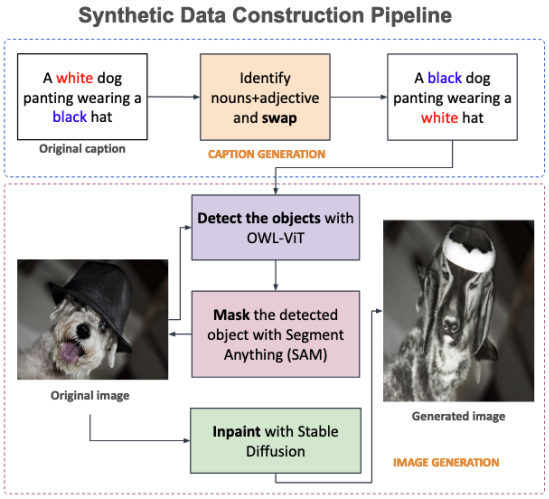

Figure 2. Overview of the synthetic data generation approach

process that involves the detection of specific objects in the augmented caption, segmenting the detected objects, and inpainting only the desired objects that were segmented. The advantage of this is twofold: **[1]** Direct image-editing with models such as plug-and-play (Tumanyan et al., 2022) fall flat with compositional prompts. Hence, we break down the compositional sentence and provide singular compositional 'chunks' to the model, only modifying the objects present. Given the original image I_o and augmented caption C_{aug}, we perform text-guided detection with OWL-ViT (Minderer et al., 2022) to detect an object that closely resembles a given adjective and noun combination (a.k.a a 'noun chunk') in the augmented caption. Next, conditioned on detected object bounding boxes, we mask the detected objects with Segment Anything(SAM) (Kirillov et al., 2023), and finally, we perform inpainting with Stable Diffusion (Rombach et al., 2022) to keep the overall scene. We perform inpainting separately for each object, breaking down the compositional step. **[2]** As a result, the rest of the augmented image (such as the background and other details) also maintains similarity with the original image.

The overall proposed data generation pipeline is represented in Figure 2.

4.2. RQ2: Enhancing Phrase-Grounding Capabilities of VLMs

Phrase-grounding models like Grounding DINO, extract and ground entities or noun phrases in the text caption to a region in the corresponding image. We see in our analysis that their performance is strongly tied to the performance of the baseline models that we explored. As observed in Figure 3, the group score for instances which exhibit highest phrase-grounding scores is the highest across all other slices. This is a strong indication that instances which can successfully ground phrases in text to objects in the image, find it easier to perform the task.

We hypothesise that:

(a) **Adding textual meta information to image captions will help VL text encoders**: Recent works (Kamath et al., 2023) have shown that text encoders in VL models are slightly worse at text understanding capabilities than standalone language encoders. We hypothesise that explicitly feeding textual meta-data such as noun phrases, adjectives, subject-verb-object, etc information along with image captions will help bridge some of the parsing capabilities lacking in VL text encoders.

(b) **Early fusion multimodal interactions will be more beneficial for models to learn compositional reasoning**: Based on our analysis, we see that ability to ground phrases is strongly linked to performing better on compositional reasoning task. We hypothesise that the early fusion exhibited in phrase grounding models like Grounding DINO will help improve compositional reasoning in pre-trained vision language models. We propose upweighting image-text matching probabilities of VL models like BLIP-2 with confidence scores of detected and grounded phrases. Further, jointly training or finetuning VL models with a phrase-grounding objective might help with learning both early fusion and late fusion multimodal interactions.

4.3. RQ3: Multi-step Visual Programming with Image Interactions

As highlighted in our analysis, model performances significantly drop with longer, compositional captions. For example, while the overall group score for BLIP2-COCO is **23%**, it drops to **3%** on the slice of samples with longest captions and increases to **41%** on the slice of samples with shortest captions (both slices contain 10% of the data). Recently neuro-symbolic approaches have started to gain popularity for visio-linguistic compositional reasoning as detailed in the §2. Typically, these are modular systems that leverage a LLM to breakdown a complex caption/question/statement about an image into multiple atomic steps, like in a program. This program is built upon several off-the-shelf image modules for object detection, cropping, question-answering, segmentation and other image manipulation functionalities. Once the program is generated, an interpreter executes this program by calling and running inference on off-the-shelf modules as specified by the program.

We find two key limitations of past works in this regard: **a) One-step program generation:** Usually, these systems take as input the whole caption, and generate a program in one step, without accessing the image during the program-generation process. The propensity to error increases with

longer captions, since the program has to lay out multiple reasoning paths in one execution. **b) No image interaction in the reasoning process:** Typically programs are generated by taking only the text caption as input. However, when we we do human simulation to solve this task, we often look at the image during each step of the reasoning process, and that informs the next step in reasoning. For example, if the caption says, *a brown dog is on a white couch*, we would first read *brown dog* and look at the image to see if that information is true. If the dog itself is not brown, we don't need to parse the rest of the sentence before concluding that the caption does not belong to the image.

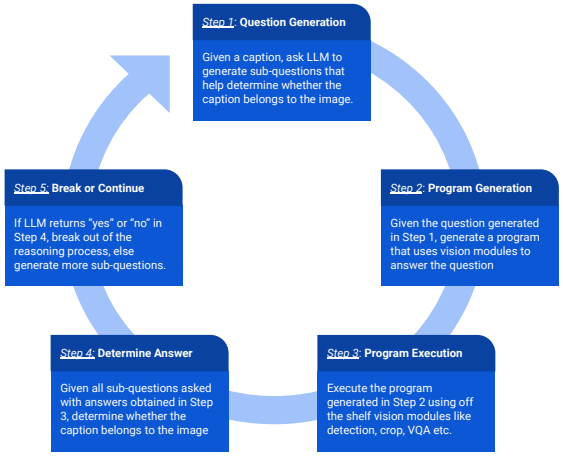

Figure 3. Overview of the proposed approach in §4.3

Therefore, in our current proposal, we aim to reason in a multi-step, sub-modular fashion, where we enable image interaction in every sub-module. Concretely, our proposed pipeline is as follows:

- *Step 1: Question Generation* - Feed in the caption to a LLM, and ask it to generate a modular sub-question which will help determine whether the caption belongs to the image. If the caption is *a brown dog is on a white couch*, we ask the LLM to generate a question of the type: *Is the color of the dog brown?*
- *Step 2: Program Generation* - Next, we generate a visual program (Gupta & Kembhavi, 2023) which leverages object detectors, segmentation maps, and other modules that can help answer the above question.
- *Step 3: Program Execution* - Third, we execute the sub-program generated in *Step 2*, to answer the question generated in *Step 1* (Is the color of the dog brown?).
- *Step 4: Answer Determination* - Now, we again prompt a LLM with the original caption (*a brown dog is on a white couch*), the generated question in Step 1 (*Is the color of the dog brown?*), and the answer obtained in *Step 3* (yes or no). We ask the LLM to decide whether it has sufficient information to answer if the caption belongs to the image. If it is certain, it should respond with *yes/no*, and if not, it is should respond with *maybe*.
- *Step 5: Break or Continue* - If the LLM responds with *yes* or *no* above, we stop here and break out of the loop with the label. For example, if the answer to the the question *Is the color of the dog brown?* is *no* after program execution in *Step 3*, we don't need more reasoning steps. However, if the LLM responds with *maybe* in *Step 4*, we loop back to *Step 1* with the question-answer pairs generated thus far and continue the reasoning process until the LLM outputs a *yes* or *no* in *Step 4*.

5. Experimental Setup

5.1. RQ1: Parallel Synthetic Data Creation for Robustness

Following the synthetic data generation approach stated in 4.1, we create a dataset of hard image-text pairs from the 2014 training subset of COCO Captions. COCO Captions 2014 contains over 590,000 images and 5 captions for each image. After applying the generation pipeline to COCO and filtering out irrelevant samples, we retain with roughy 2000 image-text pairs post augmentation. For caption augmentations, we keep captions whose similarity is above 0.97 compared to original caption. The caption similarity is computed with Sentence Transformer(MiniLM). Our analysis indicated that sentences that show higher similarity scores also showed higher semantic similarities.

We then fine-tune BLIP-2 and CLIP on the generated dataset as representative models from our baselines. For better comparison with the baselines, we use the base BLIP-2 finetuned on COCO captions (BLIP-2-COCO) and CLIP(b-32). For both BLIP-2 and CLIP, we train the model on their pretraining objective i.e. using the Image-Text matching loss, Image-text contrastive loss, and the caption generative loss for BLIP-2; we use standard Image-Text Contrastive loss(ITC) for CLIP. Image-text pre-training work such as BLIP-2 shows that in-batch negative samples improves the model robustness. Hence, we use both the original and augmented image-text pairs in the same batch as illustrated in figure 4 to enforce the 'difficult negatives' case.

In Table 1, we provide the finetuning hyper-parameters for both BLIP-2 and CLIP.

5.2. RQ2: Enhancing multimodal interactions via explicit phrase-grounding

Exp 1: Add textual meta-data to captions to explicitly to help text encoders In our analysis so far, we have

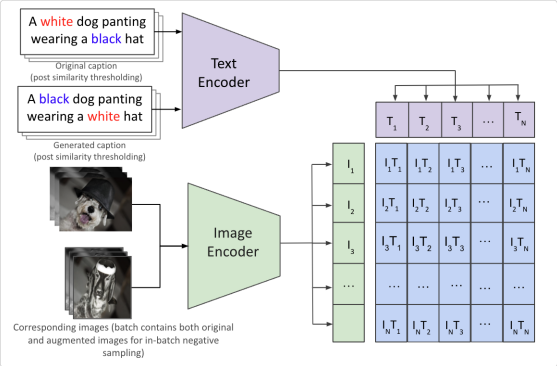

Figure 4. The overall fine-tuning workflow for CLIP. During training, both original and augmented image-text pairs are feed to the model

Hyperparams	BLIP-2	CLIP(ViT-B/32)
Fine-tuning epochs	10	5
Learning rate	1e-5	2e-5
Batch size	32	512
Adam β	(0.9,0.98)	(0.9,0.98)
Weight decay	0.1	0.1
Image resolution	364	224

Table 1. Hyperparameters for fine-tuning BLIP-2 and CLIP on synthetic data of hard image and text pairs

provided VL models with only image and caption pairs, without any additional meta-data. As a first step, we aim to examine the impact of incorporating explicit textual meta-data information into text encoders. We experiment with augmenting text captions with nouns, grounded phrases extracted from GroundingDino (Liu et al., 2023a), subject-verb-objects (SVOs), adjectives, verbs, constituency trees, AMR trees, and dependency trees.

To incorporate this meta-data, we append the caption with "where the nouns are {add nouns}" for nouns, "where the subjects are {add subjects}, verbs are {add verbs} and objects are {add objects}" for SVOs, and "where the objects to concentrate are {add DINO phrases}" for grounded phrases. For example, to incorporate DINO phrases, the original image caption *a brown dog is on a white couch* is transformed to *a brown dog is on a white couch where the objects to concentrate are 'brown dog' and 'white couch'*. For brevity, we experiment with only BLIP-2 model.

Exp 2: Upweighting BLIP-2's image-text-matching (ITM) probabilities with phrase grounding confidence scores The image-text-matching (ITM) head of BLIP-2 is a binary classification task where the model predicts whether an image-text pair is positive (matched) or negative (unmatched). We experiment with upweighting the logits from the classification head with the confidence scores from Grounding DINO.

$$\text{ITM-logits} = \mathrm{W} * \mathrm{Q}_{\text{out}}[1] + \alpha * \texttt{max}(\text{phrase conf. score})$$

where $\mathtt{W}$ are the weights of the linear layer, Q_{out} is the last hidden state of the output from the Q-former in BLIP-2, $[1]$ is the logit corresponding to 'positive' match. We use the most confident (or maximum) of the grounded phrase scores from GroundingDINO and we control this weight using hyperparameter α.

5.3. RQ3: Multi-step Visual Programming with Image Interactions

As elaborated in §4.3, we leverage LLMs in three steps out of five: *Step 1: Question Generation*, *Step 2: Program Generation*, and *Step 4: Answer Determination*. Below, we give a brief overview for the prompt instructions we used for each of these steps. We follow chain-of-thought prompting and ask the LLM to reason before generating the question or answers in *Step 1* and *Step 4* respectively. The prompts for *Step 3* are borrowed from prior work (Gupta & Kembhavi, 2023). All prompt examples can be found in the §A.1. Below, we show a brief example of how we design a in-context learning (ICL) example in the prompt.

Note that Visprog itself does not test on Winoground. Hence, as a baseline, we hand-design visual programs that aim to match the caption with an image. An example of such a program can be found in the following algorithms:

Algorithm 1 Sample ICL example for *Step 2 (question generation)*

Caption: They wear a watch on their left wrist and they carry a book in their right hand.
List of questions asked with answers by visual AI model: ['Question: Which wrist is the watch on? Answer: left']
Reasoning: The caption mentions that the person wears a watch on their left wrist and carries a book in their right hand. The question asked confirms that the watch is on the left wrist. However, I still don't know whether the book is in the left hand or the right hand. To decide if the caption belongs to the image, I will now ask the visual AI model where the book is.
Question: Which hand is the book in?

Algorithm 2 Sample ICL example for *Step 4 (answer determination)*

Caption: the brown dog is on the white couch
List of questions asked with answers: ['Question: Is the dog brown? Answer: yes']
Reasoning: The caption mentions that the brown dog is on the white couch. The question asked confirms that the dog is brown. However, I still don't know if the couch is white. So, I need more information to decide whether the caption belongs to the image.
Short Answer: maybe

Algorithm 3 Sample ICL example for visprog (baseline)

Question: Is the following caption true of the image?
Caption: black cat wearing a green sweater
Program:
BOX0=LOC(image=IMAGE,object='cat')
IMAGE0=CROP(image=IMAGE,box=BOX0)
ANSWER0=VQA(image=IMAGE0,question='Is the color of the cat black?')
BOX1=LOC(image=IMAGE,object='sweater')
IMAGE1=CROP(image=IMAGE,box=BOX1)
ANSWER1=VQA(image=IMAGE1,question='Is the color of the sweater green?')
ANSWER2=VQA(image=IMAGE,question='Is the cat wearing a sweater?')
ANSWER3=EVAL(expr="'yes' if ANSWER0 and ANSWER1 and ANSWER2 else 'no'")
FINAL_RESULT=RESULT(var=ANSWER3)

6. Results and Analysis

6.1. RQ1: Parallel Synthetic Data Creation for Robustness

Upon finetuning CLIP and BLIP-2 with augmented captions, we see an increase in performance across all types of scores in both BLIP2 and CLIP. More specifically, we see a larger increase in text-scores (by around 5% and 3% for BLIP2 and CLIP respectively) compared to image scores (2% and 2%), which we hypothesize is due to the nature of our images: we prioritize the distance and closeness of an image to its caption over the 'naturalness' of the image. To further test this hypothesis, we perform a qualitative analysis of the images.

Qualitative analysis of augmented images and captions: We perform a qualitative analysis on the image generation capabilities by subsampling 30 images from our dataset. We bin the images as follows:

- **Completely edited and far from the original caption:** These are images that appear natural looking, and also match the augmented caption more than the original caption. We see about 45% of our data (14 images) fall under this category.
- **Incompletely edited and far from the original caption:** These are images that have undergone a partial editing process (such as the modification of one object). However, they no longer match the original caption. We notice 40% of our data (12 images) fall in this category.
- **Unsound generations:** These are images that neither were edited to look natural, nor deviate from the original caption; or have completely degenerated. 15% (4 images) show this property. We notice that this is correlated with the probability of the caption occurring in the real world: (for instance 'green cows grazing over a black field' being unnatural in the real-world vs 'black cows grazing over a green field')

We display an example of each of these categories in the Appendix (Fig. 5). We believe that all images comprised within the first two categories are sufficient to enforce a distancing between 'harder' cases; since we focus on the contrastive aspects between the augmented images as opposed to the nature of the images themselves. We see a performance improvement despite 15% of the images being completely degenerated, hinting that further explorations on improving the image-editing pipeline can boost performance.

	CLIP	CLIP_{aug}	BLIP2coco	$\text{BLIP2coco}_{\text{aug}}$
Text-score	0.31	0.34	**0.44**	**0.49**
Image-score	0.11	0.13	**0.26**	**0.28**
Group-score	0.09	0.10	**0.23**	**0.24**

Table 2. Winoground scores for the baselines and upon hard-case training. BLIP2coco represents the baseline BLIP2 model finetuned on the coco captions set. We call our finetuned models CLIP_{aug} and $\text{BLIP2coco}_{\text{aug}}$

Effect of weighting the loss for images: We perform an additional experiment on CLIP wherein we weight our image-losses higher than our text loss to further increase our image score:

```
total_loss = k * loss_img(image_logits,gt)
+ (1-k) * loss_txt(text_logits,gt)
```

Where k is a weighting hyperparameter and gt is our ground truth.
However, this only leads to a decrease in the image and group scores of CLIP over multiple values of k. We hypothesize that this could be from the difference in the pretraining and finetuning objectives leading to training instability. However, we additionally notice that simply finetuning CLIP for a longer amount of time leads to an increase in the image-score, with a decrease in the

Model	Text Score	Image Score	Group Score
BLIP-2 (pretrain)	43.0	21.0	17.5
– add DINO phrases	**43.5**	20.25	15.75
– add nouns	41.0	16.71	14.25
– add SVO	38.0	20.25	16.25
BLIP-2 (coco)	44.0	25.75	23.5
– add DINO phrases	43.25	**26.5**	21.5
– add nouns	43.5	**26.75**	23
– add SVO	43.25	**26.25**	20.75
BLIP-2 (pretrain vitL)	42.5	19.25	15.25
– add DINO phrases	**43.0**	19.25	14.75
– add nouns	39.75	18.0	14.0
–add SVO	37.75	17.0	12.75

Table 3. Encoding textual meta-data explicitly using BLIP-2's text processor: we concatenate phrases extracted using GroundingDINO, nouns, subject-verb-object(SVO) to image captions

Model	Text Score	Image Score	Group Score
BLIP -2 (coco) WQ_{out}	44.0	25.75	23.5
+ (α=1) * max_score	43.75	**26.0**	**23.75**
+ (α=10) * max_score	38.75	**30.25**	**24**
+ (α=0.5) * max_score if max_score > 0.6	**45.25**	**26.5**	**24.25**
+ (α=10) * max_score if max_score > 0.6	38.5	**29.0**	23
+ (α=0.5) * max_score if max_score > 0.7	43.0	**26.25**	23.25
+ (α=10) * max_score if max_score > 0.7	38.75	**27.25**	21.75

Table 4. Upweighting BLIP-2's ITM with phrase grounding confidence scores. Here max_score indicates the scores of the most confident phrase grounded by GroundingDINO

Model	IMG0-CAP0(yes) IMG0-CAP1(no)	IMG1-CAP1(yes) IMG1-CAP0(no)	Group Score (all)
Visprog (baseline)	20.25	20.5	4.75
Multi-step (ours)	22.0	21.5	5.25

Table 5. Visual programming approaches to Winoground

text-score. We believe this could be the model learning to adapt from the synthetic images in the dataset, effectively transferring the properties to winoground.

6.2. RQ2: Enhancing Phrase-Grounding Capabilities of VLMs

Exp 1: Adding textual meta-data to image captions before feeding into text encoders In Table 3, we present results of incorporating instructions to explicitly pay attention to grounded phrases extracted from GroundingDino (Liu et al., 2023a), nouns, and subject-verb-objects (SVOs) into the text encoder of BLIP-2. It should be noted that we also ran experiments involving adjectives, verbs, constituency trees, AMR trees, and dependency trees. However, these text encoders are not equipped to directly parse such tree representations as text, which resulted in significantly lower performance scores.

We notice that adding grounded phrases explicitly to captions helps the best. These grounded phrases lead to improvements in text scores (in BLIP-2 pretrain and BLIP-2 pretrain vitL) as well as the image score (in BLIP-2 coco) in Table 3. This leads to confirming our hypothesis that detection and explicitly providing grounded phrases into a off-the-shelf VL text encoder helps enhance its text understanding capabilities, which helps with compositionality reasoning.

Exp 2: Unweighting BLIP-2's image-text-matching (ITM) with phrase grounding confidence scores In Table 4, we experiment with different ways of upweighting the 'positive' class logits of the ITM head of BLIP-2 model. Our findings indicate a consistent enhancement in the image matching score. This suggests that GroundingDINO is more confident in detecting grounded phrases in the correct pair (caption0/image0 and caption1/image1) than that of the incorrect pairs. This phrase grounding ability leads to improved performance on compositional reasoning.

We also experimented with upweighting the Winoground examples exclusively where the maximum confidence level of the grounded phrase exceeded 0.6/0.7. (higher than 0.7 is rare using GroundingDINO). We found that confidence score-based filtering was less beneficial. Instead, incorporating the confidence scores of grounded phrases, regardless of the score value, proved beneficial in enhancing compositional reasoning. Our hypothesis is that the improvement could be due to the combination of early fusion multimodal interactions learnt through GroundingDINO alongside the late fusion multimodal interactions learnt by the BLIP-2 model.

As a next step, we aimed to further fine-tune GroundingDINO with an additional image-text-matching head, similar to BLIP-2. This finetuning procedure was aimed at learning both early fusion and late fusion multimodal interactions explicitly. For the finetuning process, we utilized the MSCOCO dataset. However, we encountered issues due to unstable and unpredictable training losses. We intend to investigate this problem further as part of our future studies.

6.3. RQ3: Multi-step Visual Programming with Image Interactions

Overall, both the baseline and the multi-step method underperform end-to-end models. We make use of *gpt-3.5-turbo-instruct* as our LLM. Winoground has not been tested using visual programming approaches in the past. The group score is exceptionally challenging, because the program has to generate "yes" for IMG0-CAP0 and IMG1-CAP1 and "no" for IMG0-CAP1 and IMG1-CAP0. Hence, we additionally report an easier metric where the models get one image-caption pair correct.

Overall, the question generator and the answer determiner work well. Most errors can be attributed to the visprog module which is used in *Steps 2 and 3* to do VQA. Alternatively,

we can leverage SOTA VQA models out-of-the-box for this step, but that is something we did not consider in the current scope and stuck to visual programming in each module. Fine-grained observations are as below:

- **Shortening the reasoning path**: One desirable property of our method was for the model to break out early in the reasoning process. We observe this to happen in some cases; for example, for the caption *flat at the bottom and pointy on top*, the first question generated is *Is the object flat at the bottom?*, to which VQA returns "no". Hence, in *Step 4*, the answer determiner breaks out without needing to ask whether the object is pointy at the top.
- **Failure in captions that have relative phrases**: When a caption has two phrases that are being compared relative to each other, the reasoning process cannot be broken down into independent sub-modules. For example, for the caption: *the green one is fast and the one in white is comparatively slow*, the generated QA pairs are as follows: *Is the green one fast? Answer: no; Is the white one slow? Answer: no*, which fails to capture the relativity in speed amongst both of them.
- **Conjunction sentences benefit most**: Sentences with conjunctions highly benefit from such decompositions. For example, *the person is jumping while the cat is sitting* is broken down to *Is the person jumping?* and *Is the cat sitting?*
- **Binary output tendency by Visprog**: Visprog modules tend to return a lot of "yes" and "no" for non-binary output questions. For example, (*what is the color on the right?*), returns "yes" or *how many people are in the group?* returns "no".
- **Issues with question generator**: Sometimes the questions are too primitive and should focus on specific facts about the statement. For example, the QA pairs generated for *the person with hair to their shoulders has brown eyes and the other person's are blue* are: *Which person has brown eyes? Answer: man; Which person has blue eyes? Answer: man*; which is technically correct, but the question should be targeted at hair length.
- **Early breakout tendency by answer determiner**: We observe many cases where the answer determiner breaks out of the reasoning loop early without knowing all of the information in the caption. This may be an artifact of chosen ICL examples or prompt design, or simply LLM inefficiency. For example, for the caption *wearing a red jacket over blue* and QA list *Is the person wearing a red jacket? Answer: yes*; the answer determiner concludes that the caption belongs to the image, without asking whether the shirt is blue.

7. Conclusion

In this paper, we have explored the challenges and opportunities of understanding compositionality in vision-language models. We have adopted a bottom-up approach, first identifying and extracting key types of meta-data information from each modality, and then comparing and contrasting the performance of unimodal and multimodal baselines. We have also proposed three approaches to enhance the compositional reasoning capabilities of VLMs, namely: **[1]** Parallel synthetic data creation for robustness, where we generate hard image-text pairs that maintain semantic similarity with the original pairs, and fine-tune VLMs on the synthetic dataset, improving our baseline models on winoground. **[2]** Enhancing multimodal interactions via explicit phrase-grounding, where we leverage the phrase-grounding scores from Grounding DINO to upweight the image-text matching probabilities of VLMs, and also augment the text captions with textual meta-data to improve the text understanding of VLMs. And, **[3]** Multi-step visual programming with image interactions, where we leverage LLMs to break down complex captions into multiple sub-questions, generate and execute visual programs to answer them, and interact with the image at each step of the reasoning process. We have also performed qualitative and quantitative analyses to understand the strengths and limitations of our methods, and to identify future directions for research.

8. Team Member Contributions

Abhinav:

- Preliminary analysis on amr subject-object switches.
- Qualitative analysis of groups and captions for the augmented images (RQ1).
- Caption Augmentation and MSCOCO filtering for RQ1, finetuning of BLIP2.
- Writing of Section 6 for RQ1. Smoothened out Section 4 and 5.

Akhila:

- Formulation of research questions with Simran (Section 3)
- Analysing and visualization of object detection and image segmentation models
- Implementing all parts and sole ownership of RQ2 – inclusive of writing code, performing analysis and writing all the sections for RQ2 in Sections 4, 5, 6
- Visual Programming prompt development and initial exploration on Winoground

Jean:

- BLIP, BLIP-COCO, and BLIP-2 text and image scores.
- Literature survey and writing for Section 2.
- Text guided image generation generation pipeline in RQ1.
- Writing for RQ1, section 4 and 5.
- Preliminary analysis on vision(unimodal) which involved visualization and heatmaps for project dataset.

Simran:

- Formulating research questions based on midterm analysis (Section 3).
- Running CLIP fine-tuning baseline for RQ1.
- Implementing all parts and sole ownership for RQ3. This involves writing the code, designing prompts, running baselines, doing the analysis etc.
- Writing all parts for RQ3 in sections 4, 5, 6.
- Setting up the Zeno interface – upload metadata and all model performances.

References

Alayrac, J.-B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., Ring, R., Rutherford, E., Cabi, S., Han, T., Gong, Z., Samangooei, S., Monteiro, M., Menick, J., Borgeaud, S., Brock, A., Nematzadeh, A., Sharifzadeh, S., Binkowski, M., Barreira, R., Vinyals, O., Zisserman, A., and Simonyan, K. Flamingo: a visual language model for few-shot learning. *ArXiv*, abs/2204.14198, 2022.

Andreas, J., Rohrbach, M., Darrell, T., and Klein, D. Neural module networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 39–48, 2016.

Berglund, L., Tong, M., Kaufmann, M., Balesni, M., Stickland, A. C., Korbak, T., and Evans, O. The reversal curse: Llms trained on "a is b" fail to learn "b is a", 2023.

Changpinyo, S., Sharma, P. K., Ding, N., and Soricut, R. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3557–3567, 2021. URL https://api.semanticscholar.org/CorpusID:231951742.

Chen, X., Wang, X., Changpinyo, S., Piergiovanni, A. J., Padlewski, P., Salz, D. M., Goodman, S., Grycner, A., Mustafa, B., Beyer, L., Kolesnikov, A., Puigcerver, J., Ding, N., Rong, K., Akbari, H., Mishra, G., Xue, L., Thapliyal, A. V., Bradbury, J., Kuo, W., Seyedhosseini, M., Jia, C., Ayan, B. K., Riquelme, C., Steiner, A., Angelova, A., Zhai, X., Houlsby, N., and Soricut, R. Pali: A jointly-scaled multilingual language-image model. *ArXiv*, abs/2209.06794, 2022.

Chen, Y.-C., Li, L., Yu, L., Kholy, A. E., Ahmed, F., Gan, Z., Cheng, Y., and Liu, J. Uniter: Learning universal image-text representations. *ArXiv*, abs/1909.11740, 2019.

Cho, J., Zala, A., and Bansal, M. Visual programming for text-to-image generation and evaluation. *arXiv preprint arXiv:2305.15328*, 2023.

Cresswell, M. Logics and languages. 1973.

Diwan, A., Berry, L., Choi, E., Harwath, D. F., and Mahowald, K. Why is winoground hard? investigating failures in visuolinguistic compositionality. In *Conference on Empirical Methods in Natural Language Processing*, 2022. URL https://api.semanticscholar.org/CorpusID:253255481.

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

Hsieh, C.-Y., Zhang, J., Ma, Z., Kembhavi, A., and Krishna, R. Sugarcrepe: Fixing hackable benchmarks for vision-language compositionality, 2023.

Hu, R., Andreas, J., Rohrbach, M., Darrell, T., and Saenko, K. Learning to reason: End-to-end module networks for visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pp. 804–813, 2017.

Hu, Y., Hua, H., Yang, Z., Shi, W., Smith, N. A., and Luo, J. Promptcap: Prompt-guided task-aware image captioning. *arXiv preprint arXiv:2211.09699*, 2022.

Jia, C., Yang, Y., Xia, Y., Chen, Y.-T., Parekh, Z., Pham, H., Le, Q. V., Sung, Y.-H., Li, Z., and Duerig, T. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, 2021.

Johnson, J., Hariharan, B., Van Der Maaten, L., Hoffman, J., Fei-Fei, L., Lawrence Zitnick, C., and Girshick, R. Inferring and executing programs for visual reasoning. In *Proceedings of the IEEE international conference on computer vision*, pp. 2989–2998, 2017.

Kamath, A., Hessel, J., and Chang, K.-W. Text encoders are performance bottlenecks in contrastive vision-language models. *arXiv preprint arXiv:2305.14897*, 2023.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., Dollár, P., and Girshick, R. Segment anything, 2023.

Langley, P. Crafting papers on machine learning. In Langley, P. (ed.), *Proceedings of the 17th International Conference on Machine Learning (ICML 2000)*, pp. 1207–1216, Stanford, CA, 2000. Morgan Kaufmann.

Li, J., Selvaraju, R., Gotmare, A., Joty, S., Xiong, C., and Hoi, S. C. H. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34: 9694–9705, 2021.

Li, J., Li, D., Xiong, C., and Hoi, S. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pp. 12888–12900. PMLR, 2022.

Li, S., Yavuz, S., Hashimoto, K., Li, J., Niu, T., Rajani, N., Yan, X., Zhou, Y., and Xiong, C. Coco: Controllable counterfactuals for evaluating dialogue state trackers. *arXiv preprint arXiv:2010.12850*, 2020.

Liu, F., Bugliarello, E., Ponti, E. M., Reddy, S., Collier, N., and Elliott, D. Visually grounded reasoning across languages and cultures. *CoRR*, abs/2109.13238, 2021. URL https://arxiv.org/abs/2109.13238.

Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Li, C., Yang, J., Su, H., Zhu, J., et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023a.

Liu, X., Zhu, Z., Liu, H., Yuan, Y., Cui, M., Huang, Q., Liang, J., Cao, Y., Kong, Q., Plumbley, M. D., et al. Wavjourney: Compositional audio creation with large language models. *arXiv preprint arXiv:2307.14335*, 2023b.

Ma, Z., Hong, J., Gul, M. O., Gandhi, M., Gao, I., and Krishna, R. Crepe: Can vision-language foundation models reason compositionally? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10910–10921, 2023.

Minderer, M., Gritsenko, A., Stone, A., Neumann, M., Weissenborn, D., Dosovitskiy, A., Mahendran, A., Arnab, A., Dehghani, M., Shen, Z., Wang, X., Zhai, X., Kipf, T., and Houlsby, N. Simple open-vocabulary object detection with vision transformers, 2022.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 2021.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10684–10695, June 2022.

Schuhmann, C., Vencu, R., Beaumont, R., Kaczmarczyk, R., Mullis, C., Katta, A., Coombes, T., Jitsev, J., and Komatsuzaki, A. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *ArXiv*, abs/2111.02114, 2021.

Sikarwar, A., Patel, A., and Goyal, N. When can transformers ground and compose: Insights from compositional generalization benchmarks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 648–669, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.41. URL https://aclanthology.org/2022.emnlp-main.41.

Surís, D., Menon, S., and Vondrick, C. Vipergpt: Visual inference via python execution for reasoning. *arXiv preprint arXiv:2303.08128*, 2023.

Thrush, T., Jiang, R., Bartolo, M., Singh, A., Williams, A., Kiela, D., and Ross, C. Winoground: Probing vision and language models for visio-linguistic compositionality, 2022.

Tumanyan, N., Geyer, M., Bagon, S., and Dekel, T. Plug-and-play diffusion features for text-driven image-to-image translation, 2022.

Wang, Z., Yu, J., Yu, A. W., Dai, Z., Tsvetkov, Y., and Cao, Y. Simvlm: Simple visual language model pretraining with weak supervision. *ArXiv*, abs/2108.10904, 2021.

Wu, C., Yin, S., Qi, W., Wang, X., Tang, Z., and Duan, N. Visual chatgpt: Talking, drawing and editing with visual foundation models. *arXiv preprint arXiv:2303.04671*, 2023.

Yang, Z., Gan, Z., Wang, J., Hu, X., Lu, Y., Liu, Z., and Wang, L. An empirical study of gpt-3 for few-shot knowledge-based vqa. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 3081–3089, 2022.

Yi, K., Wu, J., Gan, C., Torralba, A., Kohli, P., and Tenenbaum, J. Neural-symbolic vqa: Disentangling reasoning from vision and language understanding. *Advances in neural information processing systems*, 31, 2018.

Yu, J., Wang, Z., Vasudevan, V., Yeung, L., Seyedhosseini, M., and Wu, Y. Coca: Contrastive captioners are image-text foundation models, 2022.

Yuksekgonul, M., Bianchi, F., Kalluri, P., Jurafsky, D., and Zou, J. When and why vision-language models behave like bags-of-words, and what to do about it?, 2023.

Zhou, D., Schärli, N., Hou, L., Wei, J., Scales, N., Wang, X., Schuurmans, D., Cui, C., Bousquet, O., Le, Q., and Chi, E. Least-to-most prompting enables complex reasoning in large language models, 2023.

A. Appendix

A.1. Prompt Templates

The exact prompts used for each of the steps detailed in §5.3 can be found below:

Algorithm 4 *Step 2 (question generation)* prompt with a sample ICL example

You are an AI assistant who has rich visual commonsense knowledge and strong reasoning abilities.
You will be provided with:
1. A caption for an image.
2. A list of questions with answers, that are provided by a visual AI model for the same image (note that the list can be empty). It's also noted that the answers are not entirely precise.

Your goal is:
To ask a question that will help determine whether the caption belongs to the image. Your question will be given to the visual AI model. Keep the question short and precise, and ask something which has not been asked before.

Let's think step by step.

Caption: They wear a watch on their left wrist and they carry a book in their right hand.
List of questions asked with answers by visual AI model: ['Question: Which wrist is the watch on? Answer: left']
Reasoning: The caption mentions that the person wears a watch on their left wrist and carries a book in their right hand. The question asked confirms that the watch is on the left wrist. However, I still don't know whether the book is in the left hand or the right hand. To decide if the caption belongs to the image, I will now ask the visual AI model where the book is.
Question: Which hand is the book in?

Algorithm 5 *Step 3 (program generation)* prompt with sample ICL examples

Think step by step to answer the question:
Question: Is the vehicle in the top of the image?
Program:
BOX0=LOC(image=IMAGE,object='TOP')
IMAGE0=CROP(image=IMAGE,box=BOX0)
BOX1=LOC(image=IMAGE0,object='vehicle')
ANSWER0=COUNT(box=BOX1)
ANSWER1=EVAL(expr="'yes' if ANSWER0 ¿ 0 else 'no'")
FINAL_RESULT=RESULT(var=ANSWER1)
Question: Are there trains or fences in this scene?
Program:
BOX0=LOC(image=IMAGE,object='train')
BOX1=LOC(image=IMAGE,object='fence')
ANSWER0=COUNT(box=BOX0)
ANSWER1=COUNT(box=BOX1)
ANSWER2=EVAL(expr="'yes' if ANSWER0 + ANSWER1 ¿ 0 else 'no'")
FINAL_RESULT=RESULT(var=ANSWER)
Question: Do you see bottles to the right of the wine on the left of the picture?
Program:
BOX0=LOC(image=IMAGE,object='LEFT')
IMAGE0=CROP(image=IMAGE,box=BOX0)
BOX1=LOC(image=IMAGE0,object='wine')
IMAGE1=CROP_RIGHTOF(image=IMAGE0,box=BOX1)
BOX2=LOC(image=IMAGE1,object='bottles')
ANSWER0=COUNT(box=BOX2)
ANSWER1=EVAL(expr="'yes' if ANSWER0 ¿ 0 else 'no'")
FINAL_RESULT=RESULT(var=ANSWER1)
Question: Is the street light standing behind a truck?
Program:
BOX0=LOC(image=IMAGE,object='truck')
IMAGE0=CROP_BEHIND(image=IMAGE,box=BOX0)
BOX1=LOC(image=IMAGE0,object='street light')
ANSWER0=COUNT(box=BOX1)
ANSWER1=EVAL(expr="'yes' if ANSWER0 ¿ 0 else 'no'")
FINAL_RESULT=RESULT(var=ANSWER1)

Algorithm 6 *Step 4 (answer determination)* prompt with a sample ICL example

You are an AI assistant who has rich visual commonsense knowledge and strong reasoning abilities.
You will be provided with:
1. A caption for an image.
2. A list of questions with answers, provided by a visual AI model for the same image (note that the list can be empty). It's also noted that the answers are not entirely precise.

Your goal is:
To analyze the questions and answers given, and reason whether the caption belongs to the image.

Let's think step by step.

Caption: the brown dog is on the white couch
List of questions asked with answers: ['Question: Is the dog brown? Answer: yes']
Reasoning: The caption mentions that the brown dog is on the white couch. The question asked confirms that the dog is brown. However, I still don't know if the couch is white. So, I need more information to decide whether the caption belongs to the image.
Short Answer: maybe

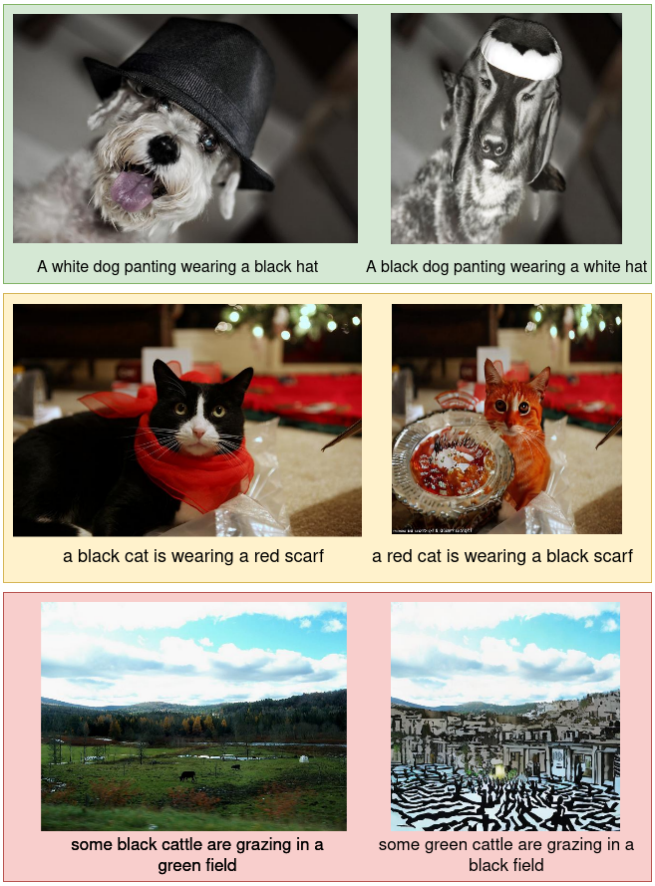

Figure 5. An example from each of the three categories we see in our augmented images; the top (in green) represents completely edited images, middle (in yellow) represents incompletely edited images, and the bottom (in red) represents unsound generations.