SEEN BUT UNNAMED: HOW CAPTION COVERAGE AFFECTS COMPOSITION IN DIFFUSION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Compositional generation asks a model to satisfy a prompt that names several objects at once, each with its attributes and relation to the scene, and diffusion models occasionally and unpredictably fail at such a task. Two kinds of accounts attempt to explain why, in the hopes of arriving at a broader claim of where compositionality originates. Architectural claims attribute composition to a property of the denoiser such as locality, under which each output pixel depends only on nearby pixels. Data-centric claims attribute it to the training data, most often to *image coverage*, the attribute combinations in the training images. However, a diffusion model also learns from its captions, and human captions leave most of the objects in an image unnamed. We introduce and vary *caption coverage*, the fraction of multi-object training images whose caption names at least two of their objects, while holding other experimental settings constant. We find that diffusion models follow a prompt that names several objects only when enough of their captions name several objects together, while image coverage mainly governs transfer to unseen color and shape pairs. We restore composition by summing a failing model's single-object predictions, and conclude that the value of captions lies in teaching a model to read several objects from one prompt. Finally, across architectural ablations, we find that cross-attention and sampler design set the caption coverage a model needs, and locality supports prompts with more objects without predicting which models trained on the same data succeed. In contrast, anyone can count caption coverage before training and raise it by rewriting captions.

1 INTRODUCTION

Composition is central to human intelligence. People who know a set of parts can place them in new relations to understand and produce combinations they have never encountered (Fodor & Pylyshyn, 1988; Lake & Baroni, 2023). Cognitive science has long studied this ability, and Fodor & Pylyshyn (1988) argued that neural networks cannot explain it. Networks that interpolate well within their training distribution often fail on familiar parts in unfamiliar combinations (Lake & Baroni, 2018; Ruis et al., 2020). Text-to-image diffusion models show both sides, combining objects and attributes in ways their training images never showed (Saharia et al., 2022) while sometimes omitting a prompted object or giving it another object's attribute (Chefer et al., 2023). The key question therefore concerns the conditions under which a network learns to generalize compositionally.

Work on this question often credits the ability to a system's design. Structured models of concept learning, for example, build concept structure into their design, as hierarchies of probabilistic concepts (Fisher, 1987) or programs that recombine parts (Lake et al., 2015). Architectural claims credit success to *locality*, under which the output at a pixel depends only on nearby pixels (Kamb & Ganguli, 2025) and on the part of the prompt about them (Bradley, 2026). Other work credits the training objective and the completeness of the conditioning (Farid et al., 2025), or the training data, and data-centric studies vary *image coverage*, the attribute combinations in the training images (Okawa et al., 2023; Uselis et al., 2025; Jeong et al., 2026). Despite this progress, these studies often vary one property in models that differ in others, so a success credited to design may come from data.

The captions offer a complementary perspective. A photograph of a cat on a couch beside a laptop shows all three objects, yet its caption may read only *a cat*. We call the fraction of multi-object images whose caption names at least two of their objects the *caption coverage* of a training set (Figure 1).

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

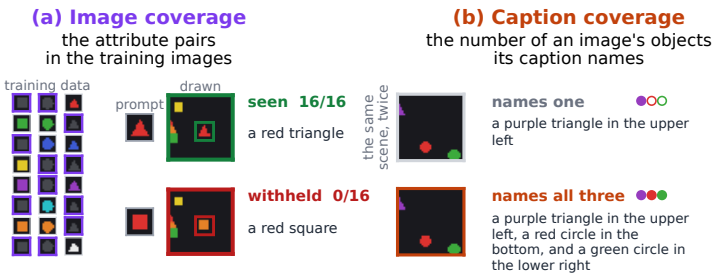


Figure 1: **Image coverage and caption coverage are distinct properties of a training set. (a)** Image coverage is the set of color and shape pairs in the training images, with withheld pairs grayed. A model trained on these images draws a seen pair in every sample and a withheld pair in none. **(b)** A caption can name one of an image's objects or all of them (dots). Caption coverage is the fraction of multi-object images whose caption names at least two of their objects.

Over the MS-COCO training images with two or more annotated categories, a human caption names 27.6% of the categories present, and only 13.9% of captions name two or more (Figure 2). However, data-centric studies take the captions as given or rewrite them whole (Betker et al., 2023; Segalis et al., 2023), and in human-captioned data a change to the images changes their captions with them. This leaves a gap between the objects a model sees and those its captions name. A fuller account should therefore vary the objects the captions name while the images stay fixed, and name each candidate cause precisely enough to measure, as Hupkes et al. (2020) did for compositionality itself.

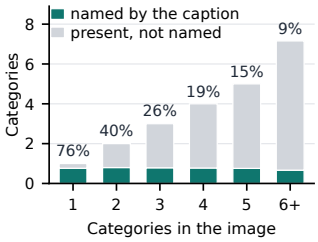


Figure 2: **Human captions name few of the objects present.** Every caption of the MS-COCO training images, by the number of categories its image holds. The colored part of a bar is the mean number the caption names, the gray part the rest, and the percentage the share named.

Diffusion models are a natural place for this test, because we can control both the network and its training data, and they can generalize compositionally (Liu et al., 2022). A diffusion model learns a *denoiser*, a network that predicts a clean image from a noisy one given a *condition*, which encodes the caption during training and a *prompt* at test time (Ho et al., 2020; Karras et al., 2022; Rombach et al., 2022). We make the captions an experimental variable and write every training caption ourselves. A caption is either a *text caption*, a sentence that names each object (embedded by a text encoder), or *object tokens*, learned vectors giving each named object's color, shape and position. We do so on SHAPEWORLD, a dataset we define, and on CLEVR (Johnson et al., 2017) and MS-COCO (Lin et al., 2014).

Contributions. We define caption coverage, a property of a training set that anyone can count before training. We show on three datasets and with two kinds of caption that raising caption coverage alone lets a model fulfill multi-object prompts, while image coverage mainly decides which color and shape pairs it can draw. A model trained at low caption coverage leaves prompted positions empty, although summing its single-object predictions fulfills the same prompts, and it recovers faster with more full captions. We replicate the link between locality and generalization to more objects (Bradley, 2026) and show that locality does not predict success among models trained on the same data.

2 RELATED WORK

Compositional generalization. Cognitive science studies the ability to form new combinations of familiar parts under the names systematicity and productivity (Fodor & Pylyshyn, 1988). Hupkes et al. (2020) split it into separately testable properties, and benchmarks probe neural networks with combinations of familiar parts that training withheld (Lake & Baroni, 2018; Ruis et al., 2020). Studies disagree on the source of the ability. Modular architectures support it in visual question answering (Bahdanau et al., 2019), properties of the environment drive it in an embodied agent (Hill et al.,

2020), and a standard network trained on many compositional tasks generalizes as people do (Lake & Baroni, 2023). We test systematicity as the recombination of colored shapes at named positions.

Diffusion models for compositional generation. Diffusion models can compose by adding the scores of separately conditioned models (Liu et al., 2022). Stable Diffusion (Rombach et al., 2022) fails on multi-object prompts in two ways, by *catastrophic neglect*, where a named object is missing, and by incorrect *attribute binding*, where an attribute goes to the wrong object (Chefer et al., 2023). Repairs steer attention during sampling (Chefer et al., 2023; Rassin et al., 2023), benchmarks measure attribute binding (Huang et al., 2023; Trusca et al., 2024) and the presence of each prompted object (Ghosh et al., 2023), and layout-conditioned generators take positions as input (Li et al., 2023). None of them varies the objects the captions name.

Architectural claims on composition. Architectural claims explain success by properties of the trained denoiser. Bradley et al. (2025) characterize when summing the scores of separately conditioned models composes them correctly, and Bradley (2026) proves that a single model conditioned on several inputs does so exactly when its output depends only locally on the image and the prompt. On CLEVR (Johnson et al., 2017), enforcing locality restores generalization to more objects (Bradley, 2026). Kamb & Ganguli (2025) derive the novelty of convolutional models from locality and equivariance. Studies compare these properties across models that differ in captions, architecture or training time, and none tests their power to predict success among identically trained models.

Data-centric claims on composition. Data-centric claims credit statistics of the training data. Okawa et al. (2023) show on colored shapes that a model generates a combination only once each of its concepts is frequent enough, which made image coverage an experimental variable. Jeong et al. (2026) vary the training images, whose condition labels follow them, and find that scene complexity matters more than concept frequency when data are scarce. In image classifiers, generalization to new combinations depends more on the variety of combinations seen than on the amount of data (Uselis et al., 2025). Denser, machine-written captions also help text-to-image models follow prompts (Betker et al., 2023; Segalis et al., 2023; Chen et al., 2024). Closest to our manipulation, conditioning on one random object per image (Bradley, 2026) or dropping factors from the conditioning (Farid et al., 2025) impairs generalization. These studies compare a few conditioning settings, change the images or rewrite captions whole, and we vary caption coverage with the images fixed.

3 Definitions and hypotheses

In this section, we define the experimental setting, the measures on which we determine effective composition, and state the hypotheses we test.

3.1 Captions, prompts and caption coverage

A training set pairs each image x with a caption c, and each object in the image, in the set $O(x)$, has a color, a shape and a position. A prompt is a caption of the same form given at test time, and a *multi-object prompt* names several objects in a combination absent from training.

We use two forms of caption. A *text caption* is a sentence, read by a frozen text encoder, that names each object by its color, shape and position, such as *a purple triangle in the upper left and a red circle in the bottom*. An *object-token caption* has one *object token* per object it names, the sum of a learned vector for its color and shape pair and one for its grid cell, both trained from scratch with the denoiser.

A *full caption* names every object in its image with everything our captions record about it, its color, shape and position on rendered scenes and its category on photographs. Any other caption is *partial*. The partial captions we build name one object, and a partial text caption ends with *among other things* so that it stays true of its image. We define the *caption coverage* of a training set D as the fraction of its multi-object images whose caption names at least two of their objects,

$$p(D) \;=\; \frac{\big|\{(x,c) \in D : |O(x)| \geq 2,\ |N(c)| \geq 2\}\big|}{\big|\{(x,c) \in D : |O(x)| \geq 2\}\big|}, \tag{1}$$

where $N(c) \subseteq O(x)$ is the set of objects that caption c names. On photographs, $O(x)$ is the set of annotated categories, and a caption names a category by containing its name. Counting every caption

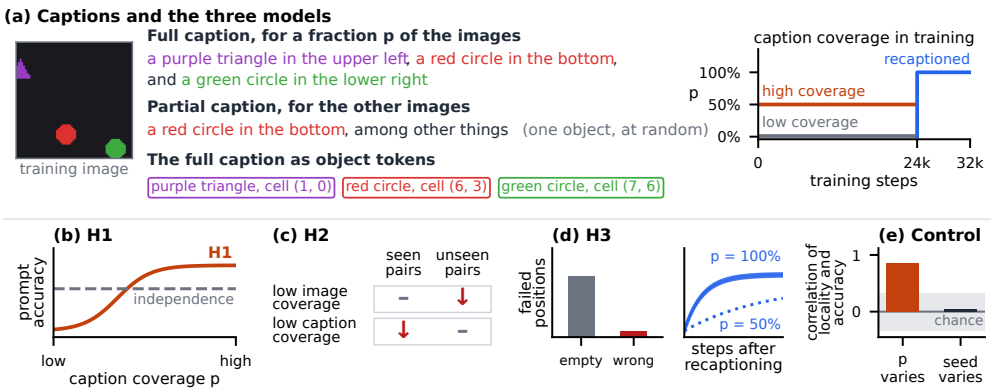


Figure 3: **The manipulation and the prediction of each hypothesis. (a)** Each training image keeps one caption, full for a fraction p of the images and partial otherwise. The plot gives each model's caption coverage over training. **(b)** H1, prompt accuracy against p, flat under the independence hypothesis. **(c)** H2, each coverage's effect on accuracy for seen and unseen pairs. **(d)** H3, empty positions, and recovery after recaptioning at $p = 100\%$ and 50%. **(e)** The control, the correlation of locality with prompt accuracy across p and across seeds, with chance in gray.

of the MS-COCO training images (Chen et al., 2015) gives $p = 13.9\%$ (Figure 2), and an image's five captions together name two or more categories for 40.0% of images (Appendix A.2).

To vary caption coverage, we keep the images fixed and write the captions ourselves. Each image receives one caption for all of training, full with probability q and otherwise partial, naming one of the image's objects at random. A caption then names each object with probability $q + (1-q)/|O(x)|$, so every color, shape and position is as frequent in the captions as in the images, and q changes only the number of objects a caption names together (Appendix A.1). The resulting set has caption coverage $p = q$ up to sampling error, and we label each set by p. *Image coverage* is the set of color and shape pairs in the training images, and we vary it by withholding pairs from every image.

On each dataset we train one model for each of several values of p and compare three main conditions (Figure 3a). We train the *low-coverage model* at the smallest p and the *high-coverage model* at a large p, on the same images. The *recaptioned model* is the low-coverage model trained further on full captions only ($p = 100\%$). It asks whether partial captions have already prepared the model, in which case it should improve quickly, and faster the longer it trained on them.

3.2 Measuring prompt accuracy

We study position as the relation between objects. A prompt names K objects. The k-th has a position ℓ_k and an attribute pair $a_k = (a_k^{\text{col}}, a_k^{\text{shp}})$, its color and its shape. On rendered scenes a position is a cell of an 8×8 grid, or one of nine regions such as *the upper left* for text captions, and on photographs, whose captions give no positions, it is the whole image. A *position classifier* r, trained on real renders, reports the attribute pair found at a position of a generated image $\hat{x}$, or $\varnothing$ for an empty position. Over test prompts with N prompted objects in total, we define

$$\text{PA} = \frac{1}{N}\sum \mathbb{1}\big[r(\hat{x}, \ell_k) = a_k\big], \qquad \text{Occ} = \frac{1}{N}\sum \mathbb{1}\big[r(\hat{x}, \ell_k) \neq \varnothing\big], \qquad \text{AM} = \frac{\text{PA}}{\text{Occ}}, \tag{2}$$

with each sum over every prompted object. *Prompt accuracy* PA is the fraction of prompted objects found at their position with their color and shape. On photographs it reduces to *presence*, whether each named object appears, which detector-based benchmarks also score (Ghosh et al., 2023). The accuracy of the position classifier on real images is the *ceiling* of prompt accuracy. *Occupancy* Occ is the fraction of prompted positions that hold any object, and *attribute match* AM is the fraction of occupied ones that hold the prompted object. A drop in occupancy matches the catastrophic neglect of Chefer et al. (2023), and a drop in attribute match their incorrect attribute binding. Each prompted object is *correct*, *misbound* when it carries another prompted object's attributes, *partial* when only its color or its shape is right, *substituted* when neither is, or *omitted* when its position is empty.

For image coverage, *competence* is prompt accuracy on color and shape pairs in the training images, and *transfer* is the fraction of that accuracy that reaches pairs withheld from every training image. In a *dissociation*, each coverage changes one of the two far more than the other.

3.3 READING THE PROMPT

The denoiser reads the prompt through cross-attention, whose softmax normalizes the weights at each position over the prompt's tokens, so they compete for attention. A *gated* denoiser replaces the softmax with an independent sigmoid gate per token. The *own share* measures how a model routes the prompt. At a prompted object's position, it is the fraction of the weight on prompted tokens that falls on its own token. It is $1/K$ for an even spread over K objects and 1 when each position attends only to its own object. *Summing* a denoiser's single-object predictions at sampling time (Liu et al., 2022) fulfills a prompt from its knowledge of each object, without reading the objects together.

Conditional locality, adapted from Bradley (2026), is the share of the change in the output, when we drop one prompted object, that falls at that object's position. *Additivity*, the *residual fraction* and the *conditional signal-to-noise ratio* measure how the objects' contributions combine (Appendix F.1).

3.4 HYPOTHESES

Our claim is that a model follows a prompt that names several objects only when enough of its captions name several objects together. Nothing in the training objective forces this, because even at $p = 0\%$ a denoiser could fulfill such a prompt. A convolutional denoiser applies the same filters at every position. It could therefore treat a prompt for a red square in the upper left and a blue circle in the lower right as two independent local tasks, each taught by captions that name one object. Under this *independence hypothesis*, prompt accuracy does not depend on caption coverage, and three hypotheses and one control test our claim against it (Figure 3b–e).

- **H1.** A model trained on multi-object images with partial captions fails multi-object prompts, and raising caption coverage alone repairs it, which rules out the images as the source.
- **H2.** The two coverages have different effects. Withholding color and shape pairs from the images lowers transfer far more than competence, and lowering caption coverage lowers competence far more than transfer, which rules out caption coverage as image coverage measured another way.
- **H3.** A low-coverage model leaves prompted positions empty, which the independence hypothesis does not predict, and recovers faster with more full captions and no head start from earlier training.
- **Control.** Denoiser properties such as locality track success across training configurations and do not predict which identically trained models succeed, which rules them out as the cause.

4 EXPERIMENTAL DESIGN

In this section, we describe the three datasets and their models, and Appendix A gives their configuration, classifier, detector and thresholds. For H2, each dataset withholds color and shape pairs, or categories, from every training image. We score each model on held-out prompts under fixed sampling noise, and train two to eight seeds per setting and twenty-four for the control at $p = 25\%$.

SHAPEWORLD. SHAPEWORLD is a world we render ourselves, with 64×64 images of one to three shapes on an 8×8 grid. Its 3 shapes and 8 colors give 24 color and shape pairs. Its images stay fixed as p varies, and it carries the tests of H1, H2 and H3. We train a U-Net (Ronneberger et al., 2015) and a diffusion transformer (DiT; Peebles & Xie, 2023), both in the EDM formulation (Karras et al., 2022), on object tokens, and a U-Net on text captions read by a frozen CLIP text encoder (Radford et al., 2021). We train the low-coverage and high-coverage models at $p = 1\%$ and 50% with object tokens and at 1% and 100% with text captions, and score 256 prompts of each size.

CLEVR. CLEVR (Johnson et al., 2017) has scenes of three to ten objects from an independent renderer, with 24 color and shape pairs of its own. We keep scenes of three to six objects, place them on the same 8×8 grid, or on a 16×16 grid, and train U-Nets on object tokens at $p = 2\%$ and 33%.

MS-COCO. MS-COCO (Lin et al., 2014) provides photographs with annotated categories. We keep the images that hold two or more categories and fine-tune Stable Diffusion (Rombach et al., 2022) at $p = 0\%$ and 100% on captions we write from those categories, such as *a photograph of a dog and a frisbee*. A detector reports whether each prompted category appears in a generated image.

5 RESULTS

In this section, we test H1, H2 and H3, find the ability a low-coverage model lacks, and test the control. Appendices B to F give further results in the same order.

5.1 MULTI-OBJECT PROMPTS NEED CAPTION COVERAGE

In all four settings, the high-coverage model fulfills three-object prompts that the low-coverage model does not (Table 1a), and the recaptioned model recovers most of the difference (Figure 4; Appendix B.1). With object tokens, accuracy on single-object prompts stays flat. Caption coverage therefore changes the ability to follow a prompt that names several objects, supporting H1. Neither more training data (Figure 6e) nor more complex scenes (Figure 7a) change the effect. At high coverage every object-token model comes within a few percent of its position classifier's accuracy, while the text-caption model stays below it and gains on single-object prompts too (Appendix A). We score MS-COCO by presence, so we rest no claim on the photographs alone. Our full captions name every object, while human captions often name some objects and omit others. We therefore also train object-token U-Nets on SHAPEWORLD in which each caption that names several objects of a three-object image names exactly two of them, which leaves caption coverage unchanged. At $p = 50\%$ these models place 0.648 of the objects of three-object prompts, against 0.932 with full captions (Table 1b; Appendix B.2). Caption coverage thus counts the captions that teach a model to follow such prompts, and the number each names sets how much it teaches. Partial captions that name every object over training, a new one each time training samples their image, do not help (Appendix B.3).

Table 1: **Caption coverage raises prompt accuracy in every setting, most when captions name every object. (a)** Three-object prompts at the low and high caption coverage of each setting, beside the ceiling. **(b)** Object-token U-Nets on SHAPEWORLD whose multi-object captions name every object (all) or exactly two (two), three seeds each.

(a)	low	high	ceiling
SHAPEWORLD, tokens	0.119	0.932	0.994
SHAPEWORLD, text	0.054	0.357	0.994
CLEVR 8×8, tokens	0.725	0.857	0.867
MS-COCO, text	0.390	0.512	0.693

(b)	two objects		three objects	
p	all	two	all	two
10%	0.450	0.336	0.276	0.190
25%	0.696	0.542	0.551	0.367
50%	0.964	0.833	0.932	0.648

5.2 THE TWO COVERAGES HAVE DIFFERENT EFFECTS

We next ask whether the two coverages build the same abilities (H2). At $p = 50\%$, we withhold 4, 8 or 12 of the 24 color and shape pairs from every training image. With 4 withheld, we vary caption coverage over $p = 1\%$, 10% and 50%, eight seeds each. Object tokens here sum separate color and shape vectors, so they can represent a withheld pair. We chose transfer's ratio form after seeing the raw values, because raw accuracy on withheld pairs cannot exceed competence (Appendix C.1).

The two coverages dissociate (Figure 5). Withholding color and shape pairs from the images leaves competence almost flat while transfer collapses, and raising caption coverage raises competence far more than transfer. In every resampling of seeds, each coverage changes one ability more than the other. Caption coverage also raises raw accuracy on unseen pairs, from 0.059 at $p = 1\%$ to 0.741 at $p = 50\%$ (Appendix C.1), so the two coverages have different effects without each acting on one ability alone. The image-coverage test replicates the collapse that Jeong et al. (2026) find as they hold out more combinations, and agrees with Okawa et al. (2023) and Uselis et al. (2025). On photographs, where we withhold annotated categories from the images, each coverage again moves one ability more than the other (Appendix C.2). On CLEVR only the image-coverage result holds,

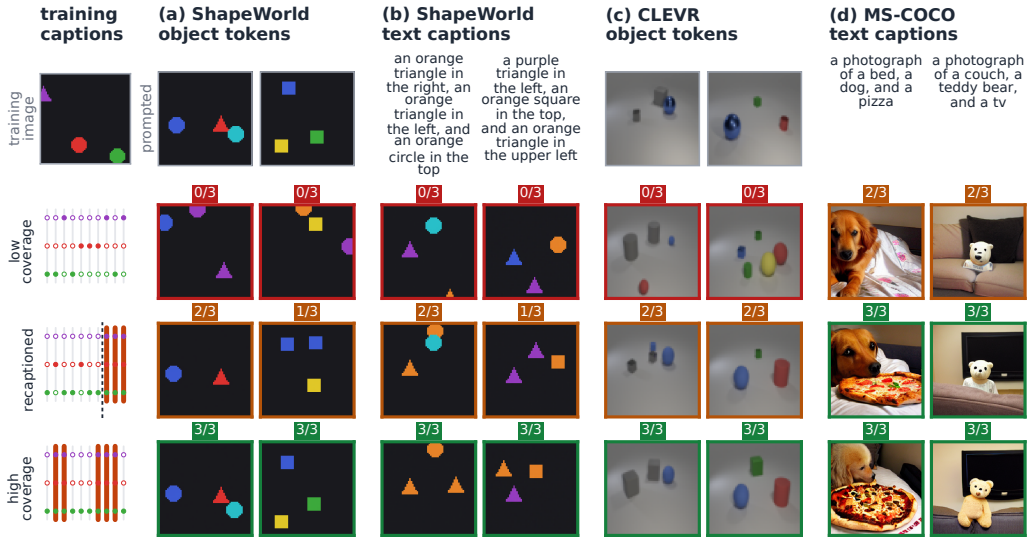


Figure 4: **The three models on the same prompts.** Columns are datasets, with prompt and noise fixed per column, and rows are models. The dots at the left show each model's training captions, one column per caption and one dot per object, filled when named and orange in full captions. The fraction above a sample counts the prompted objects placed (on MS-COCO, found by the detector).

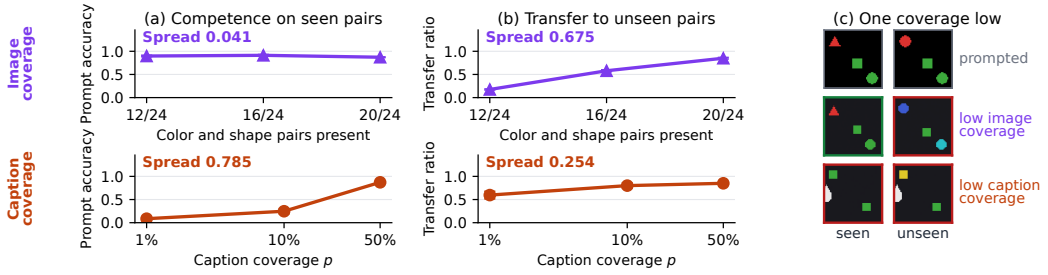


Figure 5: **Image coverage and caption coverage have different effects.** On SHAPEWORLD, image coverage is the number of the 24 color and shape pairs present. **(a)** Competence. **(b)** Transfer. Each point averages eight seeds, and the spread is the range of each curve. **(c)** A seen and an unseen pair, prompted (top) and drawn at low image coverage (middle) and at low caption coverage (bottom).

and occlusion explains why (Appendix C.3). Together, the two tests show that the images mainly supply the pairs a model can draw, and the captions mainly teach it to combine them.

5.3 LOW COVERAGE FAILS BY OMISSION

We next ask how a model without caption coverage fails and how it recovers once we complete its captions (H3). On three-object prompts, the low-coverage model leaves about 76% of its prompted positions empty, fills few with the wrong color or shape, and draws unrequested objects elsewhere (Figure 6a). A network reading each object separately would fill every prompted position, as summing its single-object predictions does (Table 2a). Caption coverage raises occupancy several times more than attribute match (Figure 6b), primarily teaching the model to place prompted objects at all. A probe locates this ability at one layer resolution of the denoiser (Appendix D.1).

Over training (Figure 6c), the high-coverage model rises within a few thousand steps, while the low-coverage model stays flat for as long as its captions name one object. Once recaptioned, it rises at a similar rate from every starting point, and a model kept at low coverage for longer reaches high accuracy a few thousand steps later (Figure 14). Training without caption coverage therefore does not

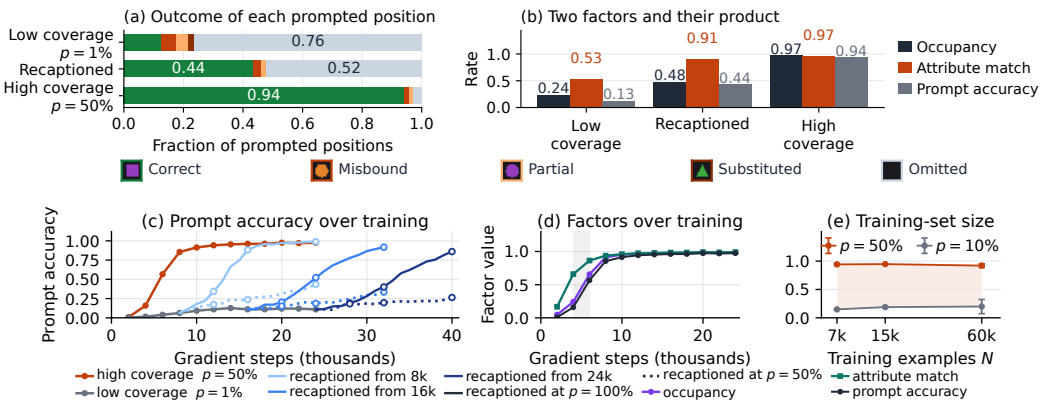


Figure 6: **Low coverage fails by omission, and training without caption coverage does not prepare the model.** All panels use SHAPEWORLD and three-object prompts. **(a)** The outcome of each prompted position and **(b)** occupancy, attribute match and their product, prompt accuracy, for the models at $p = 1\%$ and 50% and the recaptioned model. **(c)** Prompt accuracy over training at $p = 50\%$ and 1%, and for the low-coverage model recaptioned at $p = 100\%$ (solid) or 50% (dotted) from its snapshots at 8k, 16k and 24k steps. Hollow marks show where shorter budgets end. **(d)** Occupancy and attribute match of the high-coverage model, with the band where its prompt accuracy crosses one half. **(e)** Prompt accuracy by training-set size at $p = 10\%$ and 50%.

Table 2: **Low coverage fails in reading the prompt.** SHAPEWORLD U-Nets, three seeds each. **(a)** Prompt accuracy on three-object prompts. **(b)** The own share of the softmax models by layer resolution, where an even spread gives $1/3$.

(a)	$p = 1\%$	10%	50%
softmax	0.119	0.276	0.932
softmax, summed	0.951	0.951	0.966
gated	0.118	0.536	0.896

(b)	encoder			decoder		
layer	32	16	8, middle	8	16	32
$p = 1\%$	0.541	0.336	0.336	0.347	0.382	0.399
$p = 50\%$	0.547	0.335	0.338	0.447	0.597	0.519

prepare a model for it. Recaptioning at $p = 100\%$ recovers several times more than the same number of steps at $p = 50\%$ (Figure 6c), and the rise is in occupancy (Figure 6d). Eight times the data at a fixed caption coverage barely moves prompt accuracy (Figure 6e). Together, these results show that a model without caption coverage cannot read several objects from one prompt, and that more full captions teach it faster.

5.4 LOW COVERAGE FAILS IN READING THE PROMPT

Three further measurements on the SHAPEWORLD U-Nets at $p = 1\%$, 10% and 50%, three seeds each, diagnose the low-coverage model (Table 2). Summing its single-object predictions places 0.951 of the objects of three-object prompts, against 0.119 when it reads all three at once (Appendix E.1). Its captions therefore taught it each object, and it fails to read several from one prompt.

The own share locates this failure (Table 2b). We measure it on the true renders of the three-object test prompts, noised at four levels. In the decoder of the high-coverage model each prompted position gives most of its weight to its own object, while the low-coverage model spreads it almost evenly (Appendix E.2). A gated denoiser, whose tokens do not compete, fails in the same way at $p = 1\%$ and needs less caption coverage to succeed (Table 2a; Appendix E.3). Together, these results show that caption coverage teaches a network to route each prompted object to its position, and that its design sets the caption coverage it needs.

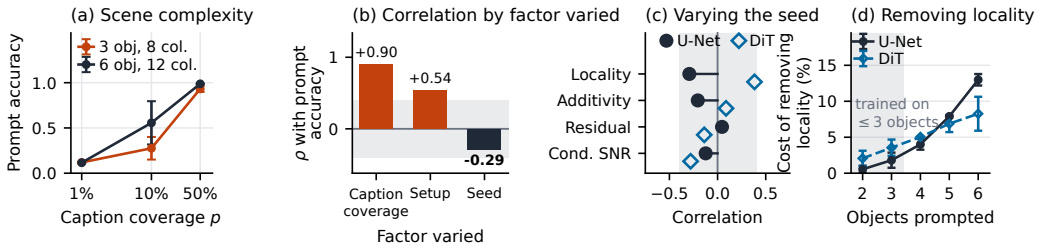


Figure 7: **Denoiser properties track prompt accuracy only when the training configuration varies. (a)** Prompt accuracy against caption coverage for scenes of up to three objects in eight colors and up to six in twelve. **(b)** Rank correlation of conditional locality with prompt accuracy by the factor varied (the setup is the caption construction, a locality penalty or the backbone). **(c)** Correlation of the four properties with prompt accuracy across twenty-four seeds at $p = 25\%$, against chance (gray). **(d)** Prompt accuracy lost to a locality penalty, for models trained on up to three objects.

5.5 Locality predicts success only across configurations

Twenty-four U-Nets trained at $p = 25\%$, differing only in their seed, cover a wide range of prompt accuracy, and crossing weight initializations with data orders traces this difference to the initialization (Appendix F.2). Longer training narrows the spread by a third (Appendix F.3). We measure conditional locality and three further properties on each model (Appendix F), and none predicts which models succeed, because every correlation with prompt accuracy lies within chance (Figure 7c). Twenty-four seeds detect a correlation of about 0.56 four times in five, so weaker links remain possible.

Locality tracks prompt accuracy across caption coverages and setups (Figure 7b), as in Bradley (2026). Removing it costs more accuracy the more objects a prompt names (Figure 7d), replicating its link to generalization to more objects. Locality therefore matters for placing many objects.

6 Conclusion and limitations

We studied how composition arises in diffusion models by naming each candidate cause precisely and varying it while the others stay fixed. Caption coverage decided whether a model follows multi-object prompts on all three datasets (H1), and image coverage mainly decided which color and shape pairs it can generate (H2). A low-coverage model left prompted positions empty and recovered faster with more full captions (H3), and locality tracked success only when the training configuration differed (the control). These findings suggest that the abilities making up composition have different sources.

Zooming out, compositionality matters because it lets finite experience cover an unbounded range of situations. A learner that composes can describe and imagine combinations it has never met, while a learner that cannot must see each combination before it can use it. The source of the ability therefore decides how to obtain it, through better architectures or better data. Our results show that captions that name one object teach each object, and that captions that name several teach the network to read them from one prompt. Gated cross-attention needs fewer such captions, and a sampler that sums single-object predictions supplies the reading by design. Overall, we argue that any claim about the source of compositionality should state the factors it holds fixed, and survive controlled changes to the design of a system and to its experience. The caption coverage we define is one such test for text-to-image models, because anyone can count it before training and raise it by rewriting captions.

Limitations. Our claims hold only in our setting, 64×64 rendered scenes of at most three objects on ShapeWorld with position as the only relation. Object tokens give each cell a learned vector, as layout-conditioned generators do, and the text-caption model, closest to deployed systems, reaches 0.357 against a ceiling of 0.994. Additionally, the photograph experiment adapts a pretrained model and scores only presence, so it may measure the erosion of an existing ability. Furthermore, the response to caption coverage differs across datasets (Appendix B.4), so it orders models only within a dataset. Lastly, the fixed-distribution control detects only strong correlations and cannot isolate the initialization's decisive property.

REPRODUCIBILITY STATEMENT

Two gate scripts run in every rebuild of this paper. The first re-derives every quantity from the raw result files and compares it with the value quoted, at the precision quoted; the second reports any numeral that no check covers, and currently reports none. Section 3 describes the caption-coverage construction and Section 4 the datasets; Appendix A.3 gives the architecture, optimizer, schedule, classifier and detector calibration, Appendix A.4 the thresholds, and Appendices B to F every further result, in the order of the results section.

We submit the anonymized source, the cluster scripts that produce every run, and the result files that both gates read as supplementary material. The script that builds the bundle runs both gates inside it and refuses to write the archive if either fails, so anyone can re-derive the reported numbers without access to our machines. The bundle omits the generated images, which its scripts re-sample.

Two caveats apply. First, a re-sample of the caption case study on a different accelerator, with the same checkpoints and the same seeded sampler and classifier, changes its per-scene counts slightly and leaves the models and their ordering unchanged. We produce every figure that quotes a per-scene count on one accelerator. Second, resampling the evaluation draw changes prompt accuracy by a small fraction of the variance across training seeds, so the spreads we report reflect training noise.

ETHICS STATEMENT

The authors have read the ICLR Code of Ethics and this work adheres to it. The study measures publicly released annotations (MS-COCO, LVIS, Flickr30k Entities, Open Images, nocaps) and fine-tunes a publicly released text-to-image model. It involves no human subjects, private data or personally identifying information, and releases no new images of people. The caption coverage we report on human-captioned datasets describes their caption sets, and we make no claim that a dataset whose captions rarely name two things is deficient in any other respect. A method for predicting when a generative model will satisfy a prompt has the dual character common to work on controllable generation. A practitioner could use the same statistic that shows how to improve prompt following to make a model follow instructions it should refuse. We believe the balance favors publication, because the quantity is a measurement of a training set and because it is at least as useful for anticipating failures of these systems as for extending them.

AI USE STATEMENT

We use generative AI tools in this work to write and refactor experiment and analysis code and to run and monitor cluster jobs. We also use them to draft and revise the text of this manuscript, to help search the literature, and as an interlocutor while interpreting results. The research ideas are the authors' own, the experiments described here measure every reported value, and code draws every figure from measured results. Experimental designs, the questions tested, the thresholds recorded in advance, and every decision to retain or withdraw a claim are the authors'.

We review all AI-assisted work. The authors verify AI-generated analysis code independently of the assistant that writes it, and check every reference found with AI assistance against its published source. The audit script described above re-derives every quantitative statement in this paper from raw result files and reports any value that a check does not cover. We take responsibility for the content of this work, including all text, claims and artifacts produced with the aid of generative AI.

REFERENCES

Harsh Agrawal, Karan Desai, Yufei Wang, Xinlei Chen, Rishabh Jain, Mark Johnson, Dhruv Batra, Devi Parikh, Stefan Lee, and Peter Anderson. nocaps: novel object captioning at scale. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8948–8957, 2019.

Dzmitry Bahdanau, Shikhar Murty, Michael Noukhovitch, Thien Huu Nguyen, Harm de Vries, and Aaron Courville. Systematic generalization: What is required and can it be learned? In *International Conference on Learning Representations*, 2019.

James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, Wesam Manassra, Prafulla Dhariwal, Casey Chu, Yunxin Jiao, and Aditya Ramesh. Improving image generation with better captions. Technical report, OpenAI, 2023. URL https://cdn.openai.com/papers/dall-e-3.pdf.

Arwen Bradley. Local mechanisms of compositional generalization in conditional diffusion. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*. PMLR, 2026.

Arwen Bradley, Preetum Nakkiran, David Berthelot, James Thornton, and Joshua M. Susskind. Mechanisms of projective composition of diffusion models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 5319–5354. PMLR, 2025.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Computer Vision – ECCV 2020*, volume 12346 of *Lecture Notes in Computer Science*, pp. 213–229. Springer, 2020.

Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. *ACM Transactions on Graphics*, 42(4):148:1–148:10, 2023.

Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. PixArt-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis. In *International Conference on Learning Representations*, 2024.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015.

Karim Farid, Rajat Sahay, Yumna Ali Alnaggar, Simon Schrodi, Volker Fischer, Cordelia Schmid, and Thomas Brox. What drives compositional generalization? The importance of continuous training objectives in visual generative models. *arXiv preprint arXiv:2510.03075*, 2025. URL https://arxiv.org/abs/2510.03075.

Douglas H. Fisher. Knowledge acquisition via incremental conceptual clustering. *Machine Learning*, 2(2):139–172, 1987.

Jerry A. Fodor and Zenon W. Pylyshyn. Connectionism and cognitive architecture: A critical analysis. *Cognition*, 28(1–2):3–71, 1988.

Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. GenEval: An object-focused framework for evaluating text-to-image alignment. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, volume 36, pp. 52132–52152. Curran Associates, Inc., 2023.

Agrim Gupta, Piotr Dollár, and Ross Girshick. LVIS: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5356–5364, 2019.

Felix Hill, Andrew K. Lampinen, Rosalia Schneider, Stephen Clark, Matthew Botvinick, James L. McClelland, and Adam Santoro. Environmental drivers of systematicity and generalization in a situated agent. In *International Conference on Learning Representations*, 2020.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. URL https://openreview.net/forum?id=qw8AKxfYbI.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851. Curran Associates, Inc., 2020.

Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2I-CompBench: A comprehensive benchmark for open-world compositional text-to-image generation. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, volume 36, pp. 78723–78747. Curran Associates, Inc., 2023.

Dieuwke Hupkes, Verna Dankers, Mathijs Mul, and Elia Bruni. Compositionality decomposed: How do neural networks generalise? *Journal of Artificial Intelligence Research*, 67:757–795, 2020.

Yujin Jeong, Arnas Uselis, Iro Laina, Seong Joon Oh, and Anna Rohrbach. When do diffusion models learn to generate multiple objects? In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*. PMLR, 2026.

Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C. Lawrence Zitnick, and Ross Girshick. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2901–2910, 2017.

Mason Kamb and Surya Ganguli. An analytic theory of creativity in convolutional diffusion models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 28795–28831. PMLR, 2025.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 26565–26577. Curran Associates, Inc., 2022.

Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1931–1941, 2023.

Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The Open Images Dataset V4: Unified image classification, object detection, and visual relationship detection at scale. *International Journal of Computer Vision*, 128(7):1956–1981, 2020.

Brenden M. Lake and Marco Baroni. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2873–2882. PMLR, 2018.

Brenden M. Lake and Marco Baroni. Human-like systematic generalization through a meta-learning neural network. *Nature*, 623(7985):115–121, 2023.

Brenden M. Lake, Ruslan Salakhutdinov, and Joshua B. Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. GLIGEN: Open-set grounded text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22511–22521, 2023.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *Computer Vision – ECCV 2014*, volume 8693 of *Lecture Notes in Computer Science*, pp. 740–755. Springer, 2014.

Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B. Tenenbaum. Compositional visual generation with composable diffusion models. In *Computer Vision – ECCV 2022*, volume 13677 of *Lecture Notes in Computer Science*, pp. 423–439. Springer, 2022.

Maya Okawa, Ekdeep Singh Lubana, Robert P. Dick, and Hidenori Tanaka. Compositional abilities emerge multiplicatively: Exploring diffusion models on a synthetic task. In *Advances in Neural Information Processing Systems*, volume 36, pp. 50173–50195. Curran Associates, Inc., 2023.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4195–4205, 2023.

Bryan A. Plummer, Liwei Wang, Chris M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k Entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 2641–2649, 2015.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021.

Royi Rassin, Eran Hirsch, Daniel Glickman, Shauli Ravfogel, Yoav Goldberg, and Gal Chechik. Linguistic binding in diffusion models: Enhancing attribute correspondence through attention map alignment. In *Advances in Neural Information Processing Systems*, volume 36, pp. 3536–3559. Curran Associates, Inc., 2023.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 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, 2022.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, volume 9351 of *Lecture Notes in Computer Science*, pp. 234–241. Springer, 2015.

Laura Ruis, Jacob Andreas, Marco Baroni, Diane Bouchacourt, and Brenden M. Lake. A benchmark for systematic generalization in grounded language understanding. In *Advances in Neural Information Processing Systems*, volume 33, pp. 19861–19872. Curran Associates, Inc., 2020.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Raphael Gontijo-Lopes, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In *Advances in Neural Information Processing Systems*, volume 35, pp. 36479–36494. Curran Associates, Inc., 2022.

Eyal Segalis, Dani Valevski, Danny Lumen, Yossi Matias, and Yaniv Leviathan. A picture is worth a thousand words: Principled recaptioning improves image generation. *arXiv preprint arXiv:2310.16656*, 2023.

Maria Mihaela Trusca, Wolf Nuyts, Jonathan Thomm, Robert Hönig, Thomas Hofmann, Tinne Tuytelaars, and Marie-Francine Moens. Object-attribute binding in text-to-image generation: Evaluation and control. *arXiv preprint arXiv:2404.13766*, 2024.

Arnas Uselis, Andrea Dittadi, and Seong Joon Oh. Does data scaling lead to visual compositional generalization? In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 60610–60634. PMLR, 2025.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014.

A Experimental setup

In this appendix, we measure the manipulation and the caption coverage of human-captioned datasets, and give the configuration and the thresholds. At high caption coverage, every object-token model trained from scratch reaches the ceiling of its own position classifier (Figure 8). The low end of each row is a separate model trained at $p = 0\%$, below the low-coverage models of Table 1, and CLEVR rises steeply between the two. Its 8×8 model reaches 0.39 of its ceiling at $p = 0\%$, 0.52 at $p = 0.1\%$, 0.67 at $p = 0.5\%$ and 0.84 at $p = 2\%$. Every CLEVR image we keep holds three to six objects, so each full caption names several objects together.

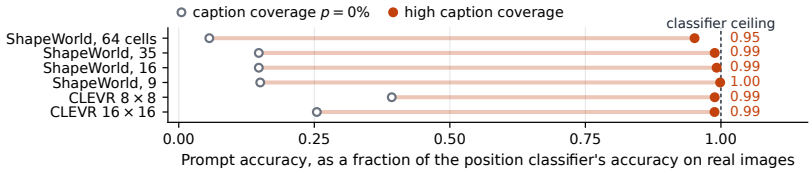


Figure 8: **Every model trained from scratch reaches the ceiling of its own position classifier.** Each row is one dataset or grid. The open mark is the prompt accuracy, the fraction of prompted objects found at their position with their color and shape, of a model trained at caption coverage $p = 0\%$, where no caption names two objects. The filled mark is the value at high caption coverage, with the number beside it, and the bar between them is the gain from caption coverage. Prompt accuracy appears as a fraction of the accuracy that the dataset's position classifier reaches on real images, so the dashed line at 1.00 is the ceiling.

A.1 Measuring the manipulation

Our construction of the captions changes the number of objects a caption names and leaves the frequency of every attribute unchanged. When each image receives a full caption with probability q, the caption of an image x names its object o with probability

$$\Pr\big[o \in C(x)\big] \;=\; q + \frac{1-q}{|O(x)|}, \tag{3}$$

which is the same for every object of the image. Among multi-object images a fraction q receive full captions, so the set has caption coverage $p = q$ (Equation 1) up to sampling error, and below we describe each set by p.

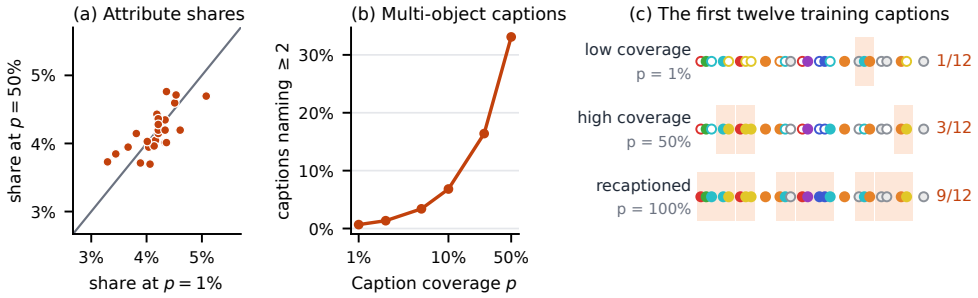


Figure 9: **Caption coverage changes the number of objects a caption names and leaves the frequency of every attribute unchanged.** Measured on ShapeWorld, where the caption coverage p is the fraction of multi-object training images whose caption names at least two of their objects. **(a)** The share of each of the 24 color and shape pairs among the named objects at $p = 1\%$ against its share at $p = 50\%$, over 4,000 scenes. **(b)** The fraction of captions that name two or more objects at each p. **(c)** The captions built for the first twelve training images of one run, at the caption coverage of each model. Each dot is one object of the image, filled when the caption names it, and a shaded cell marks a caption that names two or more objects. At $p = 1\%$ one of the twelve captions happens to name every object.

Our renderer draws an object's color, shape and position independently of the number of objects in its scene. The share of each attribute among the named objects therefore equals its share among the objects of the images at every p. Figure 9a checks this over 4,000 ShapeWorld scenes, where the shares at $p = 1\%$ and at $p = 50\%$ agree to within sampling error.

Over all captions, the fraction that name two or more objects is $p \Pr\big[|O(x)| \geq 2\big]$. The fraction is two thirds of p on ShapeWorld, whose scenes contain one to three objects with equal probability (Figure 9b), and p on CLEVR and MS-COCO, where every image contains at least two objects. At our lowest caption coverage this amounts to a few hundred such captions among the sixty thousand ShapeWorld scenes of Table 3. The left end of every response curve therefore measures

generalization from a few hundred such captions, and $p = 0\%$ is the limit at which the training objective places no constraint on multi-object prompts. Figure 9c shows the captions built for the first twelve training images of one SHAPEWORLD run.

A.2 CAPTION COVERAGE OF HUMAN-CAPTIONED DATASETS

Figure 2 compares, for every caption, the categories it names with the categories its image holds. A caption names a category when the category's name, singular or plural, appears in it as a whole word.

Over the captions of MS-COCO training images with two or more annotated categories, a caption names 27.6% of the categories present, and 13.9% of captions name two or more, against 14.3% on the validation split. On the validation images of nocaps (Agrawal et al., 2019), a caption names 26.3% of the Open Images categories present (Kuznetsova et al., 2020), a figure we qualify below.

Each MS-COCO image has five human captions (Chen et al., 2015), a few six or seven, and training usually samples among them. On the training split, one or more of an image's captions names two or more categories for 33.6% of the images, and the five captions together do so for 40.0%. The validation split gives 34.8% and 40.3%, and its first captions alone give 13.6%. A model that cycles through all five captions therefore sees several categories named together for more of the images than the first caption suggests.

Caption coverage depends on the label set we count it against. Counted against the much larger vocabulary of LVIS v0.5 (Gupta et al., 2019), the same photographs have more categories per image and a lower fraction of captions that name two or more. A larger vocabulary finds more of a photograph's contents, while the captions name no more of them. We therefore count the caption coverage of Figure 2 over the categories the detector of Appendix A.3 uses to score prompt accuracy.

Our count is also a lower bound, because it reads captions literally. Most first captions of images annotated with a person refer to the person without the word *person*, and allowing ordinary synonyms for each label roughly triples the fraction of captions that name two or more categories, mostly from *person* alone. Figure 2 therefore understates the caption coverage of human captions.

We exclude Flickr30k Entities (Plummer et al., 2015), which annotates the photographs of Flickr30k (Young et al., 2014). There an entity enters the annotation because a caption mentions it, so some caption names every object the dataset counts as present. The dataset therefore reports a caption coverage several times that of MS-COCO, which reflects the annotation protocol.

Open Images avoids this circularity, because people check the labels a classifier proposes and draw the boxes on its validation images by hand, apart from any caption (Kuznetsova et al., 2020). Two caveats keep it from being an independent test. Its labels miss more of the objects present than the annotations of MS-COCO do (Kuznetsova et al., 2020), and nocaps showed its annotators the labels of each image as keywords they need not mention (Agrawal et al., 2019). The keywords omitted the classes that Open Images lists as parts, such as a human hand or a tire, which captions rarely name. With these parts, an image holds 5.08 categories on average, against 3.43 on MS-COCO. Without them, it holds 3.72, and a caption names 30.9% of the categories present. In both datasets, a caption therefore names under a third of the objects present. Every fraction here is a mean over captions of the per-caption ratio. A ratio of the two means would differ from it where scenes vary in the number of objects they contain.

A.3 CONFIGURATION

Table 3 gives the configuration shared by every SHAPEWORLD model. Only the caption coverage, the seed and the form of the caption vary across models, and Table 1 gives the prompt accuracy of each setting at the ends of its range of caption coverage. In a text caption, the clauses follow the order in which the renderer sampled the scene's objects, which is random with respect to position and fixed for each scene. A text caption that names one of several objects marks that others are present, as in *a red square in the upper left, among other things*. It then ends with one clause *and nothing else worth mentioning* for each object it omits, so every caption of an image has the same number of clauses.

The lower ceiling of the text-caption model does not come from freezing the text encoder. An encoder trained from scratch in place of the frozen CLIP encoder lowers prompt accuracy at every caption coverage and nearly flattens the response to it. The pretrained encoder therefore lets the model

respond to caption coverage at all, and the ceiling reflects how hard it is to recover every object and its position from one sentence. With text captions, accuracy on single-object prompts also rises, from 0.21 at $p = 1\%$ to 0.86 at $p = 100\%$, so part of the text-caption model's gain in Table 1 is in reading a caption at all. At the text-caption model's lowest caption coverage, a few captions in a thousand name two or more objects, as few as for the object-token models (Table 1).

Table 3: **Configuration of the models trained on rendered scenes, and of the two scoring models.**

dataset	64×64 RGB, an 8×8 grid, 3 shapes $\times$ 8 colors (24 color and shape pairs), at most 3 objects a scene, background $(25, 25, 30)$
training set	60,000 scenes drawn per run (24,000 on CLEVR); 256 held-out prompts
denoiser	EDM-preconditioned U-Net, base 96 channels, multipliers $(1, 2, 3, 4)$, 2 residual blocks a level, cross-attention at 32, 16 and 8 and in the middle block, self-attention at 16 and 8, context 192, a learned null token
transformer model	DiT, width 384, depth 12, patch 4
caption model	frozen CLIP ViT-B/32 text encoder; learned encoder width 256, depth 4
optimizer	Adam, learning rate 2×10^{-4}, 1000 warmup steps, batch 128
schedule	24,000 steps for the sweeps, $\sigma_{\text{data}} = 0.5$, log-normal noise sampling with $P_{\text{mean}} = -1.2$, $P_{\text{std}} = 1.2$
sampling	64 deterministic steps, EMA weights
seeds	two to eight per model; twenty-four for the fixed-distribution control
position classifier	three convolutions, the last two strided, group normalization, and a linear head over the 24 pairs plus *empty*; six epochs on 8000 rendered scenes
COCO detector	DETR-ResNet-50, one threshold per category chosen by F_1 on real photographs; categories below 0.80 balanced accuracy excluded

We measure prompt accuracy on rendered scenes with the small convolutional position classifier of Table 3. We train it on rendered ground truth and never on generated images, and seed its initialization and batch order, because two classifiers trained on the same data agree on real renders yet score a generated image differently. Its accuracy on held-out renders is the ceiling for every value on rendered scenes (Figure 8).

On MS-COCO we fine-tune Stable Diffusion v1.5 at 256×256 for six thousand steps, updating only the key and value projections of its cross-attention layers, the recipe of Kumari et al. (2023). We sample with the checkpoint's default sampler for 30 steps and classifier-free guidance (Ho & Salimans, 2021) at scale 7.5. A low-coverage caption names one annotated category and marks that others are present, as in *a photograph of a dog, among other things*, and a high-coverage caption names every annotated category.

We score the generated images with DETR-ResNet-50 (Carion et al., 2020), with one threshold per category calibrated on real photographs (Table 3). This calibrates the detector against annotations, independently of the models we compare, and makes each threshold independent of the number of objects prompted. We choose each threshold to maximize F_1. Balanced accuracy tolerates a high false-positive rate for a category absent from most images, which leaves several false detections per image, while the F_1 threshold brings the number of detections per image close to the number annotated.

We exclude the categories that the detector cannot score reliably, and we measure every MS-COCO value against the ceiling the detector reaches on real photographs (Figure 8). The comparison omits the base model without fine-tuning, because we train and sample every model below the native resolution of Stable Diffusion, and its low score reflects that mismatch. The detector's false-positive rate is not negligible, so we check every verdict in Figure 4d against the detector's box.

Some prompts do not test placement. In a prompt for {keyboard, laptop}, a laptop has a keyboard, so drawing one draws both. We measure this confound from the annotations, as the larger of the two conditional probabilities that one category is present given the other, which is one when neither occurs without the other.

The confound is common, it grows with prompt size, and `person` is the category most often involved. A gap that grows with the number of prompted objects could therefore come from this confound alone, so we report trends across prompt sizes only after removing these prompts, and once we remove them no trend remains.

A.4 THRESHOLDS

We set each threshold in Table 4 from a property of the measurement, so rerunning the study would give the same value. We apply every threshold in the same way to the manipulated and the control conditions, so no threshold can create a difference between them. We set the locality threshold at the point where a measured locality lies within the spread of its own permutation null, so a model at that value is indistinguishable from one with shuffled conditions.

Table 4: **Thresholds, and how we set each.** A dagger marks the three set after the measurements they apply to.

threshold	value	how we set the value
effect floor on prompt accuracy	0.05 absolute	seed spread of a single configuration
"no conditional locality"	$1.10\times$ permutation null	the null's own sampling spread
retention floor† (an effect over the original effect)	0.30	below this the two effects are not comparable
conditional-SNR floor†	10^{-6} of prediction norm	float32 resolution of the difference
accuracy on one-object prompts†	$\geq 5\times$ chance	chance rate of the dataset
seed variance usable for the fixed-distribution control	range ≥ 0.15, s.d. ≥ 0.04	the spread the correlation needs to be estimable

We set three thresholds after the measurements they apply to, and Table 4 marks them with a dagger. We added the conditional-SNR floor after a locality measurement fell below one where the true value is exactly one, because of float32 roundoff. We added the retention floor after some ratios of two effects turned out to come from near-zero denominators. We restated the floor on accuracy on one-object prompts as a multiple of chance after a second dataset showed that an absolute value does not transfer. Chance differs by an order of magnitude across our datasets, so an absolute value would be lenient on one dataset and impossible on another.

B MULTI-OBJECT PROMPTS

In this appendix, we show further prompts for the three models, vary the number of objects a caption names and how often it changes, and compare the response to caption coverage across datasets.

B.1 CASE STUDIES

Each panel of Figure 4 claims only the verdicts its classifier supports. A position classifier whose error per position is negligible scores panels **(a)** and **(b)**. The CLEVR position classifier scores panel **(c)**. It is less accurate on real images and reads each prompted cell separately. An object drawn one cell away from its prompted position therefore counts as missing. The detector, the weakest of the three, scores panel **(d)** and misses some drawn objects that it would find in a photograph. Its panels therefore use only categories with few false positives and few false negatives, and we check every verdict against the box the detector draws. On MS-COCO the low-coverage and high-coverage models of the case studies are the first seeds of the pair of Table 1. The recaptioned model continues the low-coverage model for two thousand steps on captions that name every category.

Figures 10, 11 and 12 show further prompts on SHAPEWORLD, photographs and CLEVR, none of them in Figure 4, and they show the pattern that the aggregate numbers report. The low-coverage model leaves most prompted positions empty and draws objects that the prompt does not name, so it fails on occupancy (Figure 6). The recaptioned model fills some of the prompted positions, and the high-coverage model places every prompted object and nothing else. On SHAPEWORLD the recaptioned model places one or two of three objects, and on CLEVR it places one object fewer than the high-coverage model. On MS-COCO photographs it finds the category the low-coverage model misses in Figure 4d and none of those in Figure 11. Over the whole pool of $4{,}032$ photographs, the

detector finds 0.355 of the prompted categories for the low-coverage model, 0.401 for the recaptioned model and 0.413 for the high-coverage model.

Human captions rarely name these categories together. Over the MS-COCO training images that contain all three prompted categories, the first caption names two or more of them in 25% of cases and all three in 4%.

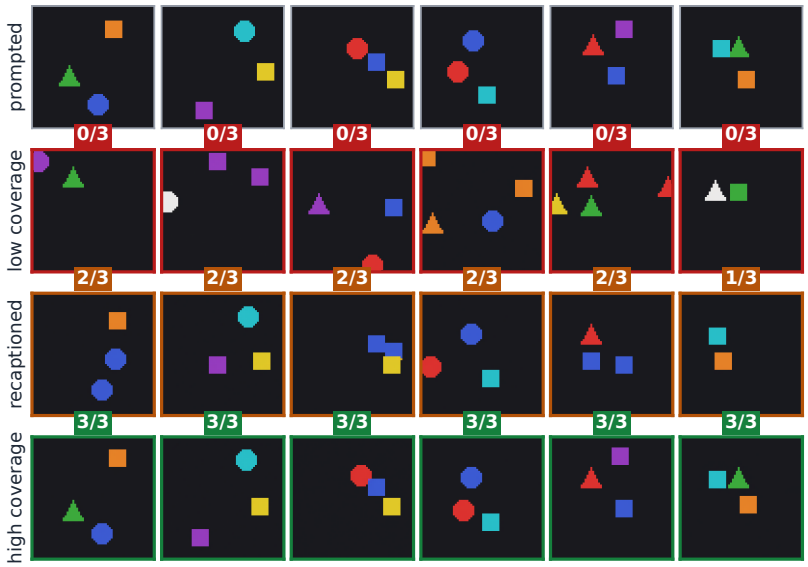


Figure 10: **Further SHAPEWORLD prompts for the three models.** Six held-out prompts, none in Figure 4, in the format of that figure. The rows are the prompted scene, the low-coverage model trained at caption coverage $p = 1\%$, the recaptioned model, which continues it for 8k steps at $p = 100\%$, and the high-coverage model at $p = 50\%$, under fixed sampling noise. Caption coverage p is the fraction of multi-object training images whose caption names at least two of their objects. The fraction above each sample counts the prompted objects it places, and the border is green when it places all of them. The low-coverage model places none and draws objects that the prompt does not name, the recaptioned model places one or two, and the high-coverage model places all three.



Figure 11: **Further MS-COCO prompts for the three models.** Four prompts from the same pool as Figure 4d, none in that figure, in its format. We fine-tune the low-coverage model on captions that name one category of each image, continue it on captions that name every category to obtain the recaptioned model, and fine-tune the high-coverage model on such captions. The fraction above each image counts the named categories that the detector finds, and the border is green when it finds all three. The low-coverage and recaptioned models find two of the three categories, and the high-coverage model finds all three.

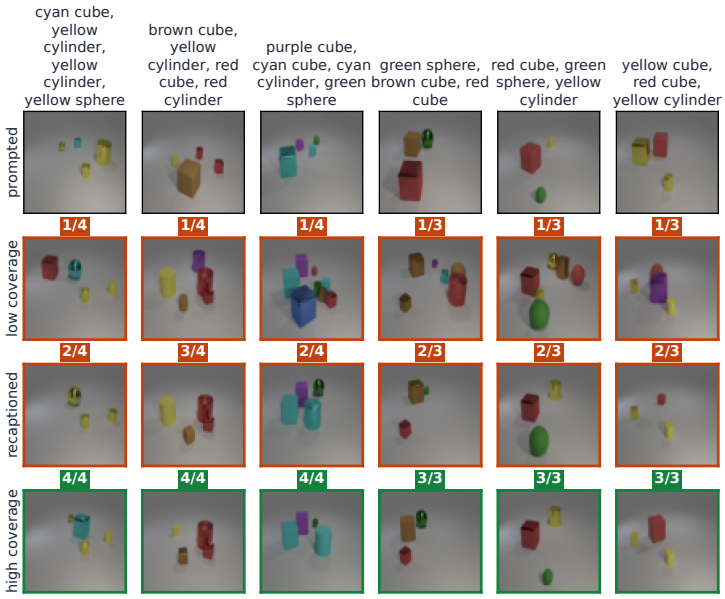


Figure 12: **Further CLEVR prompts for the three models.** Six validation prompts, none in Figure 4, in its format. The top row shows the scene each prompt comes from, with the prompted color and shape pairs above it. The rows below are the low-coverage model at caption coverage $p = 2\%$, the recaptioned model after 8k further steps at $p = 100\%$, and the high-coverage model at $p = 33\%$, under fixed noise. Caption coverage p is the fraction of multi-object training images whose caption names at least two of their objects. The fraction above each sample counts the prompted objects that the position classifier finds at their cells. The low-coverage model draws objects the prompt does not name, the recaptioned model places all but one, and the high-coverage model draws only the prompted objects.

B.2 Captions that name two of three objects

Our partial captions name one object and our full captions name every object, so every caption counted by Equation 1 in our sets names all of its image's objects. Human captions often name some objects and omit others. We therefore train models, three seeds at each caption coverage, whose counted captions name exactly two objects, chosen at random. A two-object image still receives a full caption, and p is unchanged.

Captions that name two of three objects raise prompt accuracy well above the low-coverage model, and less than full captions at the same caption coverage (Table 1b). The difference grows with p and with the number of objects a prompt names. At $p = 50\%$, three-object prompts reach 0.648 when each counted caption names two objects and 0.932 when it names every object. Equation 1 therefore counts the captions that teach a model to read several objects, and the number of objects each of them names sets how well they teach it.

B.3 Cycled captions

Human-captioned datasets give each image several captions, and training samples among them. We therefore train softmax models whose partial captions name a new object, drawn at random, each time training samples their image, so that the captions name every object of an image over training while no partial caption names two. Cycling leaves accuracy on three-object prompts (Table 5) at 0.120 at $p = 1\%$ and 0.305 at $p = 10\%$, against 0.119 and 0.276 with one fixed caption per image. Naming every object of an image over training therefore does not replace naming several objects in one caption.

Table 5: **Cycled captions still need caption coverage.** Prompt accuracy on SHAPEWORLD, the fraction of prompted objects found at their position with their color and shape, for single-object and three-object prompts. Caption coverage p is the fraction of multi-object training images whose caption names at least two of their objects. Softmax U-Nets keep one caption per image, or their partial captions name a new object each time training samples the image. Each value averages three seeds.

captions	objects	$p = 1\%$	$p = 10\%$
one caption per image	1	0.974	0.970
	3	0.119	0.276
partial captions cycled	1	0.969	0.956
	3	0.120	0.305

B.4 THE RESPONSE ACROSS DATASETS

The slope of the response to caption coverage follows a law across the grid sizes of SHAPEWORLD, where the slope is proportional to chance occupancy to within a few percent under leave-one-out. The law does not transfer across datasets.

A constant fitted on four SHAPEWORLD grid sizes predicts a slope for the 16×16 CLEVR model that the fitted slope exceeds by nearly a factor of two. That lies outside the acceptance band we fixed in advance, so we state the law for SHAPEWORLD only. No single shared exponent fits all seven models, so Figure 8 shows each model's own response, and no statistic of the training set predicts where each model reaches half its gain.

C THE TWO COVERAGES

In this appendix, we define transfer and explain the photograph and CLEVR results.

C.1 TRANSFER

The factorized object tokens of Figure 5 remove an obstacle in the encoding. With one embedding per color and shape pair, accuracy on unseen pairs stays far below chance at every caption and image coverage, because a pair absent from training has an embedding that the optimizer never updates. We instead embed color and shape separately and sum the two embeddings, so an unseen pair is a sum of embeddings that training updated many times, and its accuracy rises to many times chance.

We can state the dissociation of Figure 5 as a ratio, each coverage's change in its own ability divided by its change in the other. These ratios have wide intervals when we resample the eight seeds, but every interval excludes one, and going from four seeds to eight narrows the interval for caption coverage most. We exclude one seed at the lowest caption coverage from the resampling at that coverage, because its competence is too close to chance for a ratio to be meaningful.

Both ratios divide by a small change, so we rest the dissociation on the spreads in Figure 5, the range of each curve. The dissociation is sharper when we measure it from a floor where competence is above chance. With the caption-coverage spread taken from $p = 10\%$, caption coverage changes its own ability, relative to the other, about four times as much as with the spread from $p = 1\%$. We quote the wider spread in the main text because it is the one the design samples.

We report transfer as the ratio of unseen-pair to seen-pair accuracy, and we chose this normalization after the raw numbers were available. Our stated prediction was that raw unseen-pair accuracy is flat in caption coverage. It is not flat, and it rises from 0.059 at $p = 1\%$ to 0.741 at $p = 50\%$ over the seeds whose competence is above chance.

Competence, however, bounds the raw quantity from above. A model trained at the lowest caption coverage scores poorly even on the pairs it saw in training, so its low unseen-pair accuracy reflects its low competence, and the ratio removes this bound. The gap that caption coverage produces in transfer narrows as we add seeds, so the additional seeds move the evidence toward the dissociation.

C.2 IMAGE COVERAGE ON PHOTOGRAPHS

On MS-COCO, image coverage withholds annotated categories from the training images. We drop every image annotated with a withheld category and cut every training set to the $11{,}890$ images of the smallest pool, so the number of images stays fixed. With full captions, we withhold 0, 4, 8 or 12 categories. With 8 withheld, we vary caption coverage over the images that remain. We score three-category prompts by presence, splitting them into prompts that name only seen categories and prompts that name a withheld one.

The two coverages dissociate on photographs (Table 6). Withholding four categories lowers accuracy on prompts that name them from 0.728, when the training images show them, to 0.644, and withholding eight or twelve lowers it to 0.552 and 0.558. Accuracy on seen categories meanwhile moves only from 0.453 to 0.498.

Raising caption coverage from 0% to 100% raises accuracy on seen categories from 0.377 to 0.476, while accuracy on withheld categories moves from 0.514 to 0.552, within the spread across seeds. We chose the withheld categories for how reliably the detector finds them, so their column starts above the other.

Table 6: **The dissociation on photographs.** Presence on three-category MS-COCO prompts, the fraction of prompted categories the detector finds, for prompts that name only seen categories and prompts that name a withheld one. We withhold categories from the training images and caption every image fully, or vary caption coverage p, the fraction of multi-object training images whose caption names at least two of their objects, with eight categories withheld. With none withheld, the second column scores the same four categories, seen in training. Each value averages three seeds.

training set	seen categories	withheld categories
categories withheld from the images, $p = 100\%$		
none	0.453	0.728
4	0.462	0.644
8	0.476	0.552
12	0.498	0.558
caption coverage, eight categories withheld		
$p = 0\%$	0.377	0.514
$p = 13.6\%$	0.394	0.543
$p = 100\%$	0.476	0.552

C.3 OCCLUSION AND THE CLEVR RESULT

Occlusion explains the CLEVR result. On CLEVR, the image-coverage half of the dissociation replicates, because withholding pairs changes transfer far more than competence. The caption-coverage half does not, because caption coverage changes transfer about twice as much as competence. The 16×16 grid, which we added to give competence more room to vary, does not reverse this result in raw spreads.

We test four explanations and reject three. The first is the ceiling of the position classifier, which we reject because the pattern holds at every prompt size, including six-object prompts, where competence stays below the classifier's accuracy. The second is chance occupancy, the chance that an unrelated scene has an object in a prompted cell, which we reject because a SHAPEWORLD with nine cells, matched to the chance occupancy of CLEVR, keeps the ordering of SHAPEWORLD. The third is scene density, which we reject because a six-object SHAPEWORLD keeps the same ordering. Both control worlds keep the image-coverage half.

The fourth explanation is occlusion, and it holds. SHAPEWORLD already has some occlusion, because an object is wider than its cell, and the dissociation replicates there. When we enlarge the objects until the fraction of overlapping objects nearly triples, with everything else as in Figure 5, the pattern of CLEVR appears on our own images. Caption coverage then changes transfer more than competence, while image coverage still changes transfer far more than competence. On real renders of that world the position classifier stays well above the one-object accuracy floor of Table 4, so occlusion separates the two datasets.

One confound remains. A larger object covers more pixels, and competence in that world stays near chance at every caption coverage, so the world is harder in more than one respect. Dividing each half of the dissociation by its room, the distance from its low-coverage value to its classifier's ceiling, restores the predicted ordering on CLEVR. A low-coverage CLEVR model already scores far higher on seen pairs than a low-coverage SHAPEWORLD model, so the two halves start with very different room. We chose this normalization after the raw numbers were available (Appendix C.1), so the main text claims the dissociation in raw spreads on SHAPEWORLD and photographs, and only its image-coverage half on CLEVR.

D OMISSION AND RECOVERY

In this appendix, we refine the outcomes of Figure 6, locate the depth at which the prompt acts and report the recaptioning runs.

D.1 OCCUPANCY, ATTRIBUTE MATCH, AND DEPTH

Figure 6a sorts each prompted object into five outcomes. Misbound objects are the binding errors of Huang et al. (2023) and Trusca et al. (2024), omitted objects are the catastrophic neglect of Chefer et al. (2023), and partial and substituted objects split the remaining errors. With $\hat{a} = r(\hat{x}, \ell_k)$ the pair at a prompted object's position, the object is *omitted* when $\hat{a} = \varnothing$, *correct* when $\hat{a} = a_k$, and *misbound* when $\hat{a} = a_j$ for another prompted object $j \neq k$. Otherwise it is *partial* when $\hat{a}$ has exactly one of the color a_k^{col} and the shape a_k^{shp}, and *substituted* when it has neither.

Occupancy and attribute match multiply back to the prompt accuracy measured directly on all three models (Figure 6b), as the identity requires. Attribute match counts only occupied positions. At the low occupancy of the low-coverage model it rests on about a quarter of the prompted positions, and some of those hold objects that the prompt does not ask for. The split is therefore weakest at the low-coverage end, where the effect of caption coverage is largest, so we report the two factors beside the prompt accuracy they multiply to.

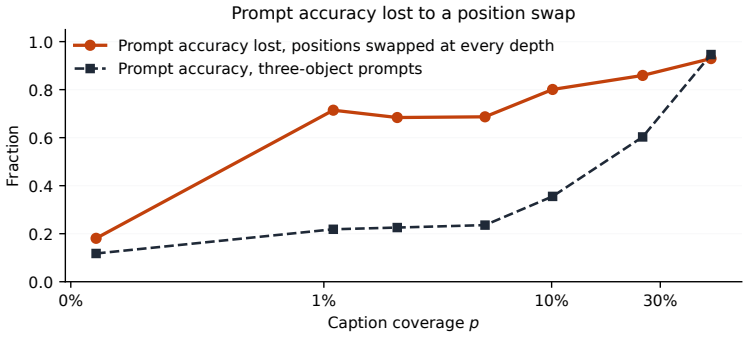


Figure 13: **Caption coverage builds the response a position swap disrupts.** On SHAPEWORLD, the fraction of prompt accuracy a model loses when we swap the positions of the prompted objects at every depth of the denoiser at once, against caption coverage p, beside its own prompt accuracy on three-object prompts. Caption coverage p is the fraction of multi-object training images whose caption names at least two of their objects. Prompt accuracy is the fraction of prompted objects drawn at their position with their color and shape.

We locate where in the network the prompt acts with a probe. The probe swaps the positions of the prompted objects in the prompt that one depth of the denoiser receives, while every other depth receives the true prompt, and we measure the prompt accuracy lost to this local swap. The probe locates a single depth on the U-Net, on the DiT and on GLIGEN (Li et al., 2023) (Table 7), so the depth at which the prompt acts is not particular to SHAPEWORLD or to its denoiser.

Figure 13 plots, across caption coverage, the fraction of prompt accuracy a model loses when the probe swaps the positions at every depth at once, beside its own prompt accuracy. At the lowest

Table 7: **The prompt acts at one depth of each denoiser.** Prompt accuracy, the fraction of prompted objects found at their position with their color and shape, when we swap the positions of the prompted objects at one depth of the denoiser. Every other depth receives the true prompt. We train the U-Net and the DiT on SHAPEWORLD with full captions and score them on three-object prompts. GLIGEN is the pretrained layout-conditioned model, scored on four-object layouts by a CLIP classifier that reads each box.

denoiser	no swap	swap at the decisive depth	lowest at any other depth	swap at every depth
U-Net	0.991	0.045 (resolution 16)	0.991	0.043
DiT	0.989	0.131 (block 3 of 12)	0.927	0.042
GLIGEN	0.661	0.107 (resolution 16)	0.661	0.102

caption coverage the probe finds little to disrupt, and the loss grows with caption coverage, so caption coverage builds the response that the probe disrupts.

D.2 RECAPTIONING

Training without caption coverage does not prepare a model for it. The recaptioning experiment has three seeds per condition, and the two distributions do not overlap, because the worst recaptioned seed exceeds the best seed trained for the same steps without recaptioning. Three seeds per side allow only twenty distinct splits, too few for a permutation test to reach significance, so we report that the two distributions do not overlap.

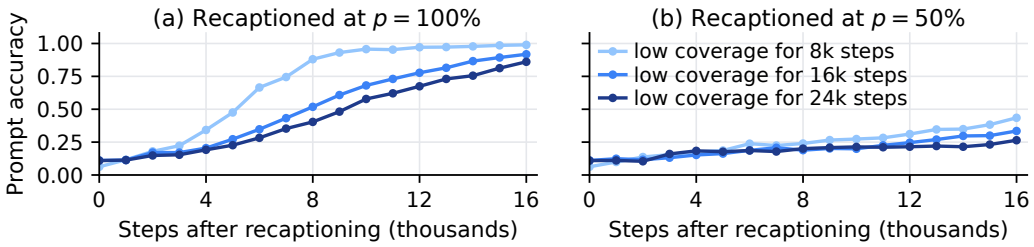


Figure 14: **Training without caption coverage does not prepare the model.** Prompt accuracy, the fraction of prompted objects drawn at their position with their color and shape, against steps from the start of recaptioning. Each run continues the low-coverage SHAPEWORLD model ($p = 1\%$) from its snapshot at 8k, 16k or 24k steps on training sets with caption coverage $p = 100\%$ **(a)** or $p = 50\%$ **(b)**. Caption coverage p is the fraction of multi-object training images whose caption names at least two of their objects. The three budgets of a starting point share a seed and coincide where they overlap.

We then test whether the steps a model spends without caption coverage help it once recaptioning begins. We resume the low-coverage run from its own snapshots at three points and continue each for three budgets at two caption coverages. Our prediction, fixed in advance, was that the final prompt accuracy depends only on the number of recaptioning steps and on the caption coverage. Figure 14 plots every recaptioned run against the steps from the start of recaptioning.

At $p = 100\%$ the three curves rise at similar rates and reach similar levels over the full budget. The model kept at low coverage for longest crosses one half a few thousand steps later than the model kept there for least. At $p = 50\%$ every curve rises slowly. Our acceptance criterion, that the final prompt accuracy agree across starting points at every budget, holds at $p = 50\%$ and at the longest budget and fails at the intermediate one, so the prediction fails as stated.

The figure nevertheless shows the effect the prediction set out to test. Training without caption coverage neither prepares a model to place several objects nor lowers the number of recaptioning steps it later needs, and if anything it delays them.

E READING THE PROMPT

In this appendix, we give the details of the further controls. Each experiment uses the models, prompts, sampler and position classifier of the main text unless we state otherwise.

E.1 SUMMING SINGLE-OBJECT PREDICTIONS

A low-coverage model may fail multi-object prompts because its captions never taught it enough to place several objects, or because it cannot read a prompt that names several. Score composition (Liu et al., 2022) separates the two. Liu et al. (2022) train their CLEVR models on the position of one random object per image and compose several objects this way. It samples with

$$D(x;\sigma,c_1,\dots,c_K) \;=\; D(x;\sigma,\varnothing) + w\sum_{k=1}^{K}\big[D(x;\sigma,c_k) - D(x;\sigma,\varnothing)\big], \tag{4}$$

where $D(x;\sigma,c_k)$ is the denoiser's prediction under a prompt that names object k alone, $D(x;\sigma,\varnothing)$ its prediction under the empty prompt, and w a weight.

Table 8: **Summing single-object predictions fulfills the prompts a low-coverage model fails.** Prompt accuracy of the SHAPEWORLD U-Nets, the fraction of prompted objects found at their position with their color and shape. The network reads the whole prompt, or we sum its predictions under each prompted object alone with weight w (Equation 4), or it reads the empty prompt. Caption coverage p is the fraction of multi-object training images whose caption names at least two of their objects. Each value averages three seeds.

model	objects	whole prompt	summed, $w=1$	summed, $w=2$	empty prompt
low coverage, $p=1\%$	2	0.263	0.968	0.991	0.003
	3	0.125	0.951	0.979	0.002
$p=10\%$	2	0.449	0.973	0.991	0.001
	3	0.282	0.951	0.976	0.002
high coverage, $p=50\%$	2	0.967	0.983	0.992	0.001
	3	0.942	0.966	0.977	0.001

Summing fulfills the prompts that the low-coverage model fails (Table 8). With $w=1$ it places 0.951 of the objects of three-object prompts, against 0.125 when the network reads the whole prompt, and occupancy rises from 0.237 to 0.978. The high-coverage model gains little, because it already reads the whole prompt. The whole-prompt column repeats the measurement of Table 1 with a newly trained position classifier and agrees with it to within 0.01.

E.2 HOW THE PROMPT REACHES EACH POSITION

The denoiser reads the prompt through cross-attention, which normalizes its weights over the tokens of the prompt. A network trained almost only on one-object captions may never learn to send each position's attention to the token of the object prompted there. Every condition also holds a learned null token, so a one-object caption still offers the attention two tokens.

We measure this routing directly. We noise the true render of each three-object prompt at $\sigma \in \{0.2, 0.5, 1, 2\}$ and read the weights of every cross-attention layer at each prompted object's cell. The *own share* is the fraction of the weight on the prompted tokens that goes to the object's own token. It is $1/3$ when a position spreads its weight evenly and 1 when it attends only to its own object.

The high-coverage model routes each token to its position in the decoder, and the low-coverage model does not (Table 2b). In the encoder at resolution 32 both models route partly and alike. In the decoder at resolution 16 the own share is 0.597 at high coverage and 0.382 at low coverage, close to the even spread. At each prompted position the low-coverage model thus mixes the prompted objects, and at most of these positions it draws nothing (Figure 6a).

Table 9: **Cross-attention without normalization still needs caption coverage.** Prompt accuracy on SHAPEWORLD, the fraction of prompted objects found at their position with their color and shape, for single-object and three-object prompts. Caption coverage p is the fraction of multi-object training images whose caption names at least two of their objects. Softmax U-Nets normalize cross-attention over the tokens of the prompt, and gated U-Nets give each token its own gate. Each value averages three seeds.

model	objects	$p = 1\%$	$p = 10\%$	$p = 50\%$
softmax	1	0.974	0.970	0.974
	3	0.119	0.276	0.932
gated	1	0.910	0.962	0.983
	3	0.118	0.536	0.896

E.3 CROSS-ATTENTION WITHOUT NORMALIZATION

Softmax cross-attention normalizes the weights at each position over the tokens of the prompt, so a model trained almost only on one-object captions may never learn to divide that weight among several objects. We therefore replace the softmax with an independent gate for each token, the sigmoid of its query-key product, whose value does not depend on the other tokens. We train such U-Nets at $p = 1\%$, 10% and 50%, three seeds each, with everything else unchanged.

The gated models need caption coverage as the softmax models do (Table 9). At $p = 1\%$ they place 0.118 of the objects of three-object prompts and 0.910 of single objects, against 0.896 of three-object prompts at $p = 50\%$. The failure at low caption coverage therefore does not come from the softmax. Summing the gated low-coverage model's single-object predictions places 0.907 of the objects, so this model also learned each object from its captions.

The gates do change how much caption coverage a model needs. At $p = 10\%$ the gated models place 0.536 of the objects of three-object prompts, against 0.276 for the softmax models, whose best seed reaches 0.452.

F THE FIXED-DISTRIBUTION CONTROL

In this appendix, we define the four properties of the denoiser that the fixed-distribution control measures, separate the initialization from the data order, and retrain its seeds for longer.

F.1 FOUR PROPERTIES OF THE DENOISER

The fixed-distribution control (Figure 7) finds that four properties of the denoiser do not predict prompt accuracy across models trained on the same distribution. We define each property here.

Table 10: **The four properties of the denoiser measured in the fixed-distribution control.**

property	definition
conditional locality	the fraction of $\|d_j\|^2$ that falls inside the cell of object j, divided by the fraction that an influence spread uniformly over the image would put there. A value of one is chance, and we set the threshold in Appendix A.4 against this null
additivity	$\|\sum_j d_j\| / \|f(C) - f(\varnothing)\|$, how much of the effect of the prompt the single-object contributions recover together
residual fraction	$\|r\| / \|f(C) - f(\varnothing)\|$, the share of the effect of the prompt that comes from interaction between objects
conditional SNR	$\|f(C) - f(\varnothing)\| / \|f(C)\|$, how much of the prediction the prompt accounts for at all. We compute the other three only where it clears the floor of Appendix A.4, because where the prompt contributes nothing, roundoff sets the location of its influence

Write $f(x, t, C)$ for the model's prediction under the whole prompt C, $f(x, t, \varnothing)$ for its prediction under the empty prompt, and

$$d_j(x, t) \;=\; f(x, t, C) - f(x, t, C \setminus \{j\}) \tag{5}$$

for the contribution of the prompted object j, obtained by dropping it from the prompt. The residual $r = f(x, t, C) - f(x, t, \varnothing) - \sum_j d_j$ is the part that the sum of the single-object contributions misses, the interaction between objects that the factorized conditionals of Bradley et al. (2025) rule out.

Table 10 defines the four properties from these quantities. We measure every property on held-out three-object prompts and average it over noise levels, on the same checkpoints that Figure 7c correlates with prompt accuracy.

Our conditional locality adapts the finite-difference measure of Bradley (2026), which relates the change at the object's own pixel to the change at all prompted pixels. Ours is the share of $\|d_j\|^2$ inside the object's cell, relative to a uniform spread. The equivalence of Bradley (2026) is a statement about score functions, while these properties are statistics of a trained network's finite-difference response to dropping one prompted object. A model can fail every property as measured here and still satisfy the theorem. The control therefore tests whether the properties predict success, and it makes no claim about the theorem.

F.2 Initialization and data order

We cross four weight initializations with four data seeds, which set the batch order and the draw of captions, and train all sixteen U-Nets at $p = 25\%$. The initialization explains most of the variance in prompt accuracy on three-object prompts, $\eta^2 = 0.673$, against 0.389 at the 95th percentile of a permutation null, and the probability of so large a value under the null is 0.0089. The data seed explains $\eta^2 = 0.062$. The spread across the seeds of the control therefore comes from the initialization, and we leave open which part of it decides the outcome.

F.3 Longer retraining

We test whether the seed-to-seed spread of Figure 7 reflects a difference in ability or in how far each seed has progressed through a transition. We retrain the same seeds for twice the budget, paired seed by seed, with two thresholds fixed in advance. A ratio of standard deviations near one means that the spread survives, and one near one half that it collapses. The measured ratio lies between the two.

Mean prompt accuracy rises and the spread narrows by about a third, with a moderate paired rank correlation, so the difference between seeds arises early in training. We compare the eight seeds that finish, because the cluster preempted four repeatedly, and without optimizer state in their checkpoints we cannot resume them comparably.