LOCAL MECHANISMS OF COMPOSITIONAL GENERALIZATION IN CONDITIONAL DIFFUSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Conditional diffusion models appear capable of compositional generalization, i.e., generating convincing samples for out-of-distribution combinations of conditioners, but the mechanisms underlying this ability remain unclear. To make this concrete, we study length generalization, the ability to generate images with more objects than seen during training. In a controlled CLEVR setting (Johnson et al., 2017), we find that length generalization is achievable in some cases but not others, suggesting that models only sometimes learn the underlying compositional structure. We then investigate locality as a structural mechanism for compositional generalization. Prior works proposed score locality as a mechanism for creativity in unconditional diffusion models (Kamb & Ganguli, 2024; Niedoba et al., 2024), but did not address flexible conditioning or compositional generalization. In this paper, we prove an exact equivalence between a specific compositional structure (*conditional projective composition*) (Bradley et al., 2025) and scores with sparse dependencies on both pixels and conditioners (*local conditional scores*). This theory also extends to feature-space compositionality. We validate our theory empirically: CLEVR models that succeed at length generalization exhibit local conditional scores, while those that fail do not. Furthermore, we show that a causal intervention explicitly enforcing local conditional scores restores length generalization in a previously failing model. Finally, we investigate feature-space compositionality in color-conditioned CLEVR, and find preliminary evidence of compositional structure in SDXL.

1 INTRODUCTION

Conditional diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song & Ermon, 2019; Song et al., 2020) appear to possess remarkable compositional generalization capabilities. For example, text-to-image models (Dhariwal & Nichol, 2021; Rombach et al., 2022; Ramesh et al., 2022) generate convincing images for prompts like “a photograph of a cat eating sushi with chopsticks” that were (probably) not seen during training. These models may generalize by composing known concepts (e.g. cat+sushi) in novel ways. However, the extent of generalization in large-scale models is unclear as their train sets are not publicly known (perhaps they *have* seen cats eating sushi). Further, despite recent progress (Okawa et al., 2024; Park et al., 2024; Sclocchi et al., 2025; Kadkhodaie et al., 2023; Favero et al., 2025; Chen et al., 2024; Wang et al., 2024; Lukoianov et al., 2025), the mechanisms underlying compositional generalization remain unclear.

We first propose a concrete and controlled setting in which to study compositional generalization: length generalization in location-conditioned models trained on CLEVR Johnson et al. (2017), a synthetic dataset of objects with various locations, shapes, and colors. Length generalization refers to the ability to generate more objects than seen in training – e.g., can a location-conditioned model trained on 1-3 objects and tested on $K > 3$ locations actually generate images with K objects at the correct locations? Prior work demonstrated length generalization of *explicit* composition of multiple diffusion models via linear score combination Du et al. (2023); Liu et al. (2022); Bradley et al. (2025). In contrast, we study length generalization of a *single* conditional model. By training on multi-object samples, we hope that this model can learn the underlying compositional structure of the data, hence length-generalize. We find empirically that, depending on conditioning and architecture specifics, these models sometimes length-generalize and sometimes do not.

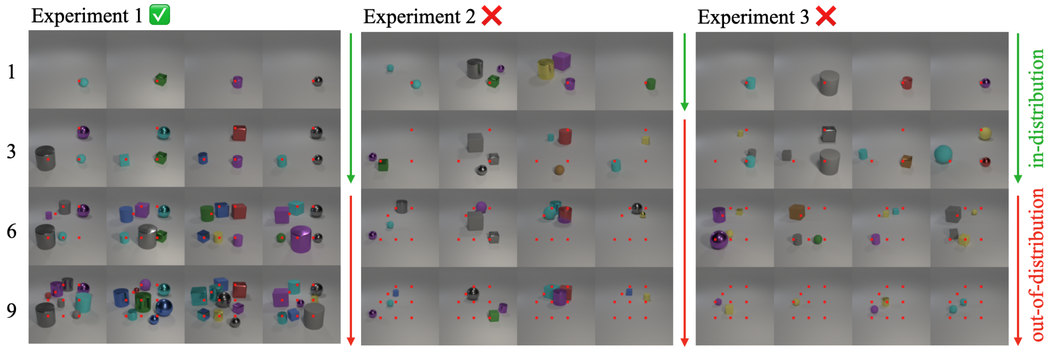

Figure 1: **Length generalization in location-conditioned CLEVR models.** We study length generalization in location-conditioned models trained on images with 1-3 objects, and tested on 1, 3, 6, 9 locations (6, 9 are OOD), with red dots indicating the conditioned locations at test-time. For each experiment, the rows correspond to different conditioners (1, 3, 6, or 9 locations) and the columns show 4 different samples. All models have the same architectures and training data and differ only in the design of their conditioners (see Figure 7). In **Experiment 1**, a grid-style conditioner labels the locations of all objects in the scene; the model successfully length-generalizes up to 9 locations. In **Experiment 2**, a grid-style conditioner labels the location of only a single object (randomly selected); the model fails to length-generalize (in this case, even 3 locations is OOD). In **Experiment 3**, a list-style conditioner labels the locations of all objects; this model fails to length-generalize beyond 3 objects. Additional samples shown in Figure 8.

Next, we study local mechanisms for compositional generalization. We build primarily on two lines of prior work: one on local mechanisms for creativity, and another on compositionality in diffusion. First, Kamb & Ganguli (2024); Niedoba et al. (2024) recently proposed that models learn *local score functions*, enabling creativity via mosaicing of local patches from different images. These works only study unconditional and class-conditional diffusion, however, and do not consider flexible conditioners such as those used in text-to-image diffusion, which are central to questions of compositional generalization. It therefore remains unclear whether local mechanisms are relevant to compositional generalization in conditional diffusion models. Second, Bradley et al. (2025) propose a formal definition, called *projective composition*, of "correct" composition of multiple distributions Du et al. (2023); Liu et al. (2022). In this paper, we specialize projective composition to a single conditional distribution to provide a precise definition of composition structure.

We develop a theoretical framework connecting compositional generalization with local mechanisms. Specifically, we generalize the concept of local scores to define *local conditional scores* (LCS): scores with *sparse dependencies* on both pixels and conditioners. That is, the score at a given pixel depends only on a subset of other pixels (such as a local neighborhood) and on only one or a few relevant conditioners (e.g. in the case of location-conditioning, only conditioners near the current pixel). We specialize projective composition (Bradley et al., 2025) to define a *conditional projective composition* (CPC) – a conditional distribution that is a projective composition of its own individual conditionals. We then prove an equivalence between conditional projective composition and local conditional scores at all noise levels. We extend this theoretical framework to relate compositional structure and sparse score dependencies in feature-space (intuitively, concepts like style+content will compose in feature-space if the score of each 'style feature' depends only on a sparse set of style-related conditioners and other features).

We validate this theory through experiments. Returning to our location-conditioned CLEVR setting and comparing a model that we found to length-generalize with others that did not, we find that the length-generalizing model maintains pixel- and conditional-locality, while the non-length-generalizing models exhibit non-locality. We find that the correlation between length-generalization and conditional-locality also holds over a wider range of models with varying length-generalization. Further, we perform a direct causal intervention to test local conditional scores as a possible mechanism for composition generalization: we show that explicitly enforcing a local architecture enables length generalization in a model that previously failed. Finally, we investigate feature-space com-

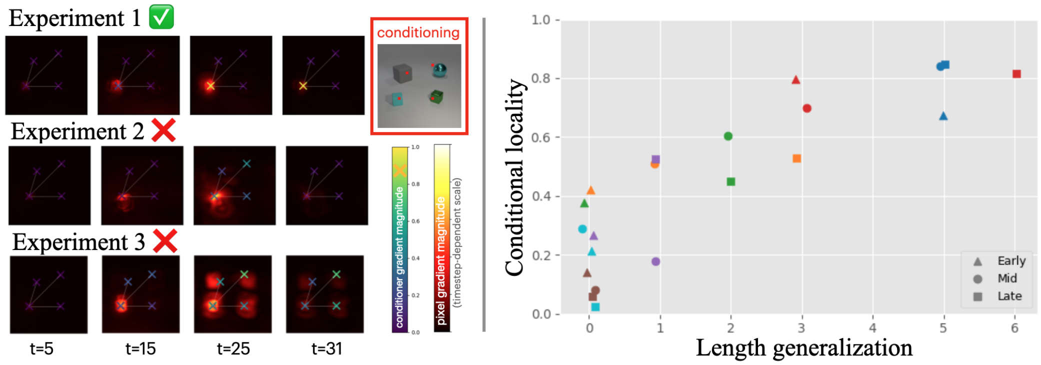

Figure 2: **Locality in location-conditioned CLEVR models** (Left) For Experiments 1, 2 and 3 of Figure 1 each conditioned on four locations, we visualize pixel-locality via heatmaps, and conditional locality via the intensity of the × marker, centered at a pixel in the lower left, over a range of timesteps. (Appendix F describes the locality measurements; Figure 13 plots locality metrics; Figure 10 shows more pixel locations.) The length-generalizing Exp. 1 model exhibits strong pixel- and conditional-locality, while the non-length-generalizing Exp. 2 and 3 models both lack conditional-locality (the scores depend on non-local conditioners); Exp. 3 also lacks pixel-locality. These experiments support the theoretical equivalence between CPC and LCS. (Right) Length generalization vs. conditioner locality for several models (different colors), each checkpointed early, mid, and late in training (different shapes). Details are in Appendix E.2.1. Length generalization and conditional locality are strongly correlated, and can emerge together over the course of training (e.g. orange, green, red models). Here, length-generalization (x-axis) is the number of locations to which the model can generalize *beyond* the number on which it was trained (e.g. +6 for a model trained on 1-3 locations that generalizes to 9). The conditional locality (y-axis) metric is described in Appendix F.

positionality in color-conditioned CLEVR, and show preliminary SDXL experiments to explore compositional structure in real-world text-to-image models.

2 Length generalization in CLEVR

Train data	Exp.1	Exp.2	Exp.3	Exp.2L	Col.
1 object	1	1	1	6	1
1-3 objects	9	1	3	9	4
1-5 objects	10	1	5	10	7

Table 1: **Upper limits of length generalization** in location- and color-conditioned CLEVR. The table lists the maximum value, $K_{\max}$, such that the model "sometimes succeeds" for every $1 \leq K \leq K_{\max}$, as described in Appendix E.2. Results are shown for Exp. 1, 2, 3 of Figure 1, Exp. 2L of Figure 3, and the Color experiment of Figure 4.

In this section we study length generalization in conditional diffusion models trained on CLEVR datasets Johnson et al. (2017), using a standard EDM2 U-net architecture Karras et al. (2022) (details in Appendix E.1). Figure 1 shows length generalization or lack thereof in three location-conditioned models trained on CLEVR images with 1-3 objects. In **Experiment 1**, the location-conditioning labels *all* objects in the scene, using a 2d integer array representing a 2d grid over the image via the count of objects whose center falls within the grid cell (typically zero or one), as shown in Figure 7. We find that this model length-generalizes up to 9 objects. In **Experiment 2**, the setup is identical to that of Experiment 1 except that the conditioning only labels the location of a *single* randomly-selected object. Unsurprisingly, this model fails to length-generalize beyond one location. **Experiment 3** conditions on the 2D locations of all objects using a list-style conditioner which places the (embedded) xy-locations of each object in an array padded with enough slots for up to 10 objects, with each location placed in a randomly chosen slot. This model fails to length-generalize beyond the 3 locations it was trained on. A priori, we should not "expect" length-generalization. Although the data is naturally compositional, there is no guarantee that a model will learn a compositional structure from examples with only 1-3 objects (which it

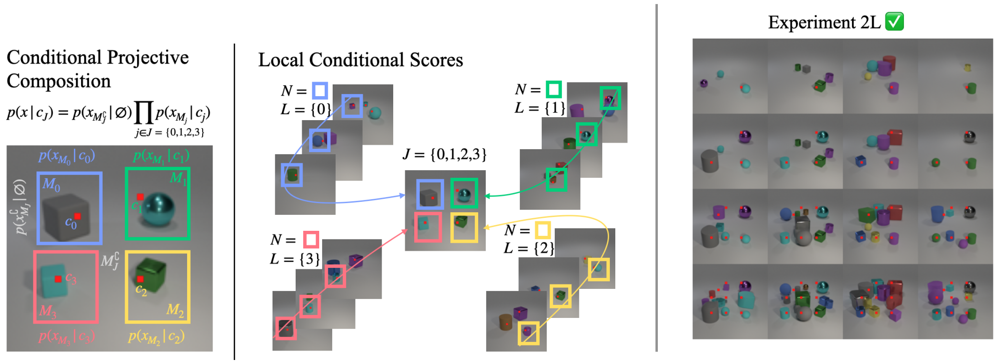

Figure 3: (Left) *Conditional projective composition* (CPC) and *local conditional scores* (LCS). A CPC is a conditional distribution over a set of conditions $c_{\mathcal{J}}$ that factorizes independently into the marginals over x_{M_j} conditioned on c_j, where M_j are disjoint subsets. An LCS is a conditional score over a set of conditions $c_{\mathcal{J}}$ such that the score at each pixel i depends only on a subset N_i of other pixels (often a local neighborhood) as well as a subset $L_i \subset \mathcal{J}$ of conditions (for location-conditioning, often nearby conditioners). For certain choices of subsets, CPC and LCS are equivalent. (Right) **Experiment 2L** applies a causal intervention to the failing Exp. 2: we modify the model architecture to explicitly enforce local conditional scores, use the same training data and conditioning as Exp. 2, and find that Exp. 2L length-generalizes while Exp. 2 failed. Locality metrics and plots for Exp. 2L are shown in Figure 13 and 10.

could fit in many different ways). Evidently, the Experiment 1 model learns the underlying compositional structure of the data while the two other models do not. Table 1 gives a quantitative analysis of the limits of length generalization of the various location- and color-conditioned models, trained on 1 to M objects for $M = 1, \ldots, 6$, with samples shown in Figure 11. The table reports an approximate measure, $K_{\max}$, of the maximum number of objects to which each model can consistently length-generalize (details in Appendix E.2).

3 Theory: Compositionality and Locality

In this section we present theory connecting compositional generalization to generalized local mechanisms. We first define *local conditional scores* (LCS), an extension of local scores Kamb & Ganguli (2024); Niedoba et al. (2024) to account for flexible conditioners. An LCS evaluated at a given pixel has sparse dependencies on other pixels (generalizing local neighborhoods) and sparse dependencies on conditioners. Next, we define *conditional projective composition* (CPC), a special case of *projective composition* (PC) proposed in Bradley et al. (2025) applied to a single conditional distribution. A conditional distribution that satisfies CPC factorizes into independent distributions over disjoint subsets of pixels that depend on a single condition. LCS and CPC are illustrated in Figure 3. We then prove that the score of a CPC is exactly an LCS: intuitively, *compositional distributions have local scores*. We verify this relationship empirically in Figure 2, discussed further in Section 4. This result can be partially relaxed in an approximation that improves at higher noise, which could allow approximately-compositional structure to be resolved early in denoising. The theory can also be extended to *feature space*, to connect compositional structure (e.g. style+content) with sparse score dependencies (e.g. scores of 'style features' only depend on select conditioners and features relevant to style). The remainder of this section makes these claims precise.

Background Our theory builds upon two lines of prior work: local scores and projective composition. Kamb & Ganguli (2024); Niedoba et al. (2024) propose that diffusion models learn local score functions, enabling creativity (generating samples not in the training set) through mosaicing of local patches. Both works define local scores essentially as follows: let x_N denote the restriction of x to a subset of indices $N \subseteq [n]$, and $p(x_N|c)$ denote the marginal distribution of $p(\cdot|c)$ on x_N. Intuitively, a local score at pixel i depends only on a neighborhood of pixels centered at pixel i. That is, s^t is a local score at time t with neighborhood subsets N_i^t (which may depend on the time t), if $s^t[x](i) := \nabla \log p^t[x_{N_i^t}](i)$, for all pixels i. However, these works study unconditional and class-conditional diffusion, not flexible conditioners central to compositional generalization. In this work, will will extend this concept to incorporate flexible conditioners and study compositionality.

Bradley et al. (2025) introduce projective composition (PC) as a formal definition for "correctly composing" multiple distributions $\{p_b, p_1, p_2, \dots\}$, where p_b is a "background" distribution and the p_i are "concept distributions". One possible construction of a projective composition is given by $p_{\mathcal{J}}(x) := p_b(x_b) \prod_{j\in\mathcal{J}} p_j(x_{M_j})$, where the M_j are disjoint subsets corresponding to the p_j, respectively. In this paper, we will specialize PC to define a specific compositional structure for conditional distributions. However, in contrast to Bradley et al. (2025)'s focus on explicit compositions, our goal is to connect compositional structure with score locality.

Next, we present new definitions and theory that build on local scores and projective composition.

Local Conditional Scores We generalize the idea of local scores (Kamb & Ganguli, 2024; Niedoba et al., 2024) to account for flexible compositional conditioners,[1] which are central to compositional generalization. Let $p(x|c)$ be the true distribution over data $x \in \mathbb{R}^n$ conditioned on c. Let x_N denote the restriction of x to a subset of indices $N \subseteq [n]$, and $p(x_N|c)$ denote the marginal distribution of $p(\cdot|c)$ on x_N. Conditioners are represented as $c_{\mathcal{J}} = \{c_j, j \in \mathcal{J}\}$, where $\mathcal{J} \subseteq \mathcal{J}_{\text{all}}$ is a subset of all possible conditioners. We assume that p is defined for any combination of conditioners, even those not seen during training. A local conditional score at pixel i, $s^t[x|c_{\mathcal{J}}](i)$, depends on two subsets (Figure 3): N_i, a subset of pixels relevant to pixel i, and $L_i(\mathcal{J})$, a subset of conditions in $\mathcal{J}$ relevant to pixel i. In general, the subsets N_i and L_i need not be disjoint (although this will later be necessary to achieve CPC equivalence), and may contain multiple objects or conditioners.

Definition 1 (Local Conditional Score (LCS)). *We say that s^t is a local conditional score at time t with pixel subsets N_i and conditional subsets L_i^t (which may both depend on the time t), if*

$$s^t[x|c_{\mathcal{J}}](i) := \nabla \log p^t[x_{N_i^t}|c_{L_i^t(\mathcal{J})}](i), \quad \textit{for all pixels } i. \tag{1}$$

Importantly, Definition 1 does not strictly require "locality" but rather captures a "sparse dependency structure" where the score at index i depends only on specific subsets N_i and L_i. While N_i is often a local neighborhood in image settings (and L_i can be local e.g. for location-conditioners), these subsets can be arbitrary in general. We use "local" as an intuitive term for these sparse dependencies.

Conditional Projective Composition To define a compositional structure for conditional distributions, we introduce conditional projective composition (CPC). We do not claim all distributions have this structure, but we will show that those that do also have a local score structure (LCS), suggesting a mechanism for compositional generalization. We specialize Bradley et al. (2025)'s pixel-space projective composition to the case where the concept distributions p_j represent the a *single* conditional distribution $p(x|c_j)$ conditioned on different c_j, and the background distribution $p_b(x)$ is $p(x|\emptyset)$ (i.e., with no conditioners active).

Definition 2 ((Pixel-space) Conditional Projective Composition (CPC)). *We say that $p(x|c)$ is a conditional projective composition if there exist disjoint sets M_j for all conditions $j \in \mathcal{J}_{all}$ such that, for any set of conditions $\mathcal{J} \in \mathcal{J}_{all}$, $p(x|c_{\mathcal{J}})$ decomposes as*

$$p(x|c_{\mathcal{J}}) := p(x_{M^{\complement}_{\mathcal{J}}}|\emptyset) \prod_{j\in\mathcal{J}} p(x_{M_j}|c_j), \tag{2}$$

where $M^{\complement}_{\mathcal{J}} := \mathbb{R}^n \setminus \cup_{j\in\mathcal{J}} M_j$ denotes the set of pixels not controlled by any active condition.

This definition means that the conditional distribution p decomposes into independent marginal distributions $p(x_{M_j}|c_j)$, each depending only on subset M_j and condition c_j, as shown in Figure 3. That is, p modifies M_j according to c_j independently of other pixel sets and conditioners. This condition is quite strong, but we will partially relax it in our theory.

3.1 Equivalence between compositional structure and local scores

In this section we present theory showing that the score of an (approximately) conditional projective composition is (approximately) a particular local conditional score. We begin by showing that a specific local conditional score is *exact* for a conditional projective composition.

[1] Our definition breaks slightly from the originals in defining local scores in terms of a distribution rather than a finite training set, and also omits equivariance, which is unnecessary for our theory.

Lemma 1 (Local conditional score is exact for conditional projective composition)**.** *Let p be a pixel-space CPC (Definition 2) with disjoint sets $\{M_j\}$. Let s^t be an LCS (Definition 1) with subsets:*

$$L_i^t(\mathcal{J}) = \begin{cases} \{j\} \cap \mathcal{J}, & \textit{if } i \in M_j \\ \emptyset, & \textit{else}, \end{cases} \qquad \textit{and} \quad N_i^t = \begin{cases} M_j, & \textit{if } i \in M_j \\ M_b, & \textit{else}, \end{cases} \qquad \textit{where } M_b := M^{\complement}_{\mathcal{J}_{all}}.$$

Then s^t is exactly the score of p^t:

$$s^t(x|c_{\mathcal{J}}) = \nabla \log p^t(x|c_{\mathcal{J}}), \quad \forall \mathcal{J}.$$

The proof is in Appendix B. The lemma says that when the locality structure of s^t is precisely connected to the compositional structure of p^t – that is, if pixel i belongs to the subset M_j controlled by condition j in the CPC, then $L_i = \{j\}$ (pixel i only depends on condition j), and $N_i = M_j$ (pixel i only depends on pixels in M_j) – then s^t is exactly the score of p^t. Thus, there is an *equivalence* between CPCs and LCSs. This is illustrated in Figure 3.

A relaxation What about imperfect compositionality? We can relax Lemma 1 to show that the score of an *approximately* CPC distribution is *approximately* an LCS. Further, we show that the CPC approximation becomes more accurate – intuitively, distributions are "more compositional" – at higher noise. The precise statements and proofs are given in Appendix C. Why might this be helpful? If conditional dependencies are strongest at high noise and pixel dependencies take over at low noise (as we observe in Figure 2), then local conditional mechanisms might be able to establish large-scale compositional structure (like object count and location) early in denoising, leaving less-compositional details to be resolved at low noise via local unconditional denoising.

3.2 Feature-space conditional projective composition

What if CPC does *not* hold in pixel-space – as is typically the case for non-location conditioners, like the color-conditioning of Figure 4, or text-to-image prompts such as "a watercolor of a cat eating sushi with chopsticks"? Pixel-space CPC is unlikely since each condition potentially affects many pixels (e.g., "watercolor" style would apply to all pixels). In these cases, we hypothesize that the local unconditional denoising mechanism still applies at low noise (Kamb & Ganguli, 2024; Niedoba et al., 2024). But is there any hope of compositional generalization at high noise?

It follows directly from Lemma 1 that if a distribution has a CPC structure *in feature-space* then its score is an LCS *in feature-space* (we name these F-CPC/F-LCS, respectively):

Corollary 1 (F-LCS is exact for F-CPC; informal)**.** *Suppose that $p(x|c)$ is an F-CPC (a CPC in feature-space): that is, $\mathcal{A}\sharp p(z|c)$ is a CPC, where $\mathcal{A}$ is an invertible transform, and $z := \mathcal{A}(x)$ is the feature-space representation. Then the feature-space score $\nabla_z \log(\mathcal{A}\sharp p)^t(z|c)$ is an F-LCS (an LCS in feature-space) with neighborhoods N_i, L_i related to the CPC subsets $\{M_j\}$ of $\mathcal{A}\sharp p$ as in Lemma 1.*

For example, in an appropriate feature-space, the concepts "watercolor", "cat", and "sushi" might have F-CPC structure – despite interacting in pixel space. (Note that an F-LCS will usually have sparse-dependencies rather than literal "locality", as allowed by Definition 1, since interacting features need not be contiguous). However, challenges remain: if the scores are F-LCS in some feature-space, in order to exploit the sparse structure the model must *learn* this feature-space mapping and its inverse, in addition to the local subsets, making the learning process significantly more challenging.[2] This argument is made precisely in Appendix D.

As a practical heuristic for identifying F-LCS structure, we propose an empirically-testable necessary-but-not-sufficient condition for F-LCS, based on orthogonality between score differences (similar to Bradley et al. (2025) Lemma 8.1). The proof is in Appendix D.1.

Lemma 2 (F-LCS necessary-but-not-sufficient heuristic)**.** *Let $s^t_{\mathcal{A}}(z|c) := \nabla_z \log(\mathcal{A}\sharp p)^t(z|c)$ be an F-LCS score in a feature-space given by transform $\mathcal{A}$, and let $t_{\max}$ denote the highest noise level. Then:*

$$d_i^T d_j = 0, \quad \forall i \neq j, \quad \textit{where } d_i := \mathbb{E}[s_{\mathcal{A}}^{t_{\max}}(\cdot|c_i)] - \mathbb{E}[s_{\mathcal{A}}^{t_{\max}}(\cdot|\emptyset)]$$

[2] In fact, even learning the feature-space transform and its inverse for the noiseless distribution is not enough – the model technically needs to learn a feature-map for every noise level. In practice it may be approximately sufficient to learn a single mapping and its inverse, though this is not entirely clear.

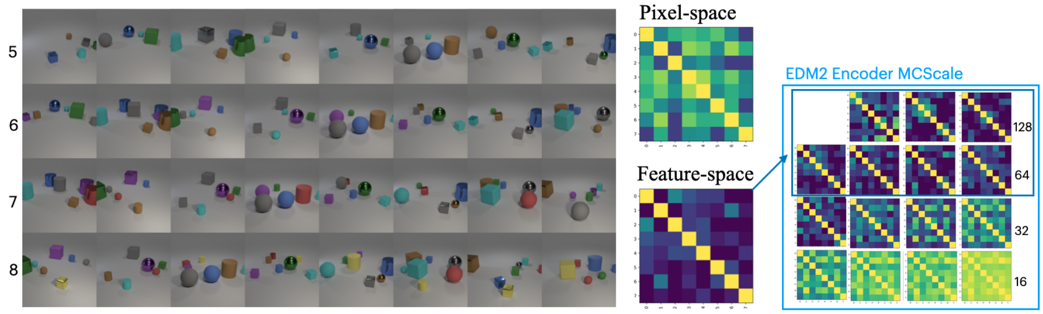

Figure 4: **Length generalization in color-conditioned CLEVR.** (Left) We test length generalization in a CLEVR model conditioned on *colors* (rather than locations), trained on 1-5 objects. We test up to 8 colors (blue, brown, cyan, gray, green, purple, red, yellow), and find generalization up to 7 colors, suggesting that compositional structure may exist in a learned feature-space. (Right) We visualize F-LCS disentanglement between colors in the network's internal representation via the Lemma 2 heuristic (low cosine similarity off-diagonal indicates F-LCS). F-LCS structure seems to appear within early encoder layer activations, suggesting that these layers might be learning a compositional "feature-space" potentially contributing to length-generalization.

where the expectation is over the feature-space transformed noise distribution $\mathcal{A}\sharp\mathcal{N}(0, \sigma_{t_{\max}})$.

Practically, to study a feature-space represented within layer ℓ of a denoising network, we compute $s_{\mathcal{A}}^{t_{\max}}(\cdot|c)$ by drawing a noise sample, running the first denoising step (at time $t_{\max}$) to compute the conditional score, and hooking the activation of layer ℓ. To obtain d we average over multiple noise samples and compute the conditional-unconditional difference. Finally, we can construct a cosine similarity matrix $\{d_i d_j / \|d_i\| \|d_j\|\}_{i,j}$: low similarity off-diagonal is evidence of F-LCS.

Remark 1. *F-CPC/F-LCS structure should be viewed as a type of* disentanglement*, on which there is a rich literature: for example, (Bengio et al., 2013; Higgins et al., 2017; Chen et al., 2018; Kim & Mnih, 2018; Locatello et al., 2019; Kotovenko et al., 2019; Locatello et al., 2019; Watters et al., 2019; Yang et al., 2023; Zhang et al., 2023) . To quote Karras et al. (2019): "There are various definitions for disentanglement, but a common goal is a latent space that consists of linear subspaces, each of which controls one factor of variation." F-CPC/F-LCS satisfies this definition with the additional requirement that the subspaces be orthogonal. Intuitively, disentanglement is often thought to promote compositionality; our specific definitions and theory of F-CPC/F-LCS makes this connection precise and provable.*

Remark 2. *Identifying feature-space disentanglement is fundamentally difficult since independence between high-dimensional random variables cannot be tested in polynomial time. However, a variety of practical metrics have been proposed (Higgins et al., 2017; Kim & Mnih, 2018; Chen et al., 2018); Locatello et al. (2019) shows that many common metrics are fairly correlated with each other. The heuristic of Lemma 2 is part of the broader family of closely-related disentanglement metrics, but is specifically designed to test F-CPC/F-LCS.*

4 Additional Experiments

Our theory shows an equivalence between LCS and CPC (which implies length generalization). We test this directly in location-conditioned CLEVR models, where the compositional structure holds in pixel-space, and location-conditioners possess a direct notion of locality. Further, we test whether LCS could be a *causal* mechanism via a direct intervention: enforcing an explicitly LCS architecture to "fix" length generalization that previously failed. Turning to feature-space compositionality, we show partial length generalization in color-conditioned CLEVR and connect it with feature-space LCS structure. Finally, we make preliminary investigations of local/compositional structure in both pixel- and feature-space in SDXL.

Pixel-space locality in location-conditioned CLEVR Figure 2 (Left) shows pixel- and conditional-locality in Experiments 1, 2, and 3. We first observe that Exp. 1 and 2 maintain pixel-locality at both low and high noise levels, contrasting with prior empirical findings on datasets like CIFAR-10 (Kamb & Ganguli, 2024; Niedoba et al., 2024) – reproduced in Figure 12 – showing delocalization at high noise. The difference likely stems from CLEVR images containing multiple, nearly-independent objects; unlike the datasets with a single centered subject studied in prior work. The non-length-generalizing Exp. 3 model lacks pixel-locality. Second, we note significant differences in conditional-locality between Exp. 1 vs. 2 and 3. The length-generalizing Exp. 1 model exhibits strong conditional-locality at high noise, while the non-length-generalizing Exp. 2 and 3 models lack conditional-locality at high noise (scores near conditioned locations either fail to respond to any conditions or depend on several non-local conditions). At low noise, all models transition to pixel-local *unconditional* denoising, as in Kamb & Ganguli (2024); Niedoba et al. (2024). These experiments support our prediction that length generalization depends on pixel- and conditional-locality, and suggest that conditional-locality at high noise plays a particularly important role. Locality metrics are plotted in Figure 10, Figure 10 shows additional pixel locations, and Appendix F details the locality measurements. Figure 2 (Right) plots length generalization vs. conditioner locality for several models (different colors), each checkpointed early, mid, and late in training (different shapes). Details are in Appendix E.2.1. Length generalization and conditional locality are strongly correlated and can emerge together over the course of training.

A Causal Intervention in location-conditioned CLEVR Experiment 2L (Figure 3) tests our theory via a direct causal intervention, wherein we design a local model architecture that explicitly enforces local conditional scores (as confirmed in Figure 10). This design is conceptually related to local model architectures proposed in prior works like Watters et al. (2019); Li et al. (2023); Zheng et al. (2023); Cheng et al. (2023). We train the local model using the same conditioning as the failing Exp. 2 (labeling only a single object location), and find that the local architectural intervention causes it to length-generalize (Table 1), "fixing" the failure. In fact, the local model trained on *only one object* can length-generalize up to 6 locations (Figure 9, Table 1). This supports the hypothesis that local conditional scores could be a *causal* mechanism for compositional generalization. Details in Appendix E.3.

Length generalization in color-conditioned CLEVR To better understand feature-space compositionality, we explore length generalization in color-conditioned CLEVR in Figure 4 and Table 1. For color-conditioning, we might expect compositional structure to exist in an appropriate feature space. We observe that length generalization is possible to some extent – e.g. a model trained on 1-5 objects can generalize to 7 colors. Next, we investigate whether the model actually learns a L-FCS disentangled feature-space by analyzing layer activations using the heuristic Lemma 2 heuristic, and find evidence of L-FCS within several layers. Details in Appendix E.5.

A preliminary SDXL investigation Can local mechanisms help to explain compositional generalization in real-world diffusion text-to-image models? While length generalization served as a controlled and verifiable special-case of compositional generalization in our CLEVR experiments, we now return to the broader question of compositions of novel combinations of concepts. We begin to explore this question by studying compositional/local structure in both pixel- and feature-space in a pretrained ('out-of-the-box') SDXL model (Podell et al., 2023) model. In Figure 5, we investigate pixel-space locality, by choosing a prompt that contains implicit location information ("middle", "right", "left"). We find some degree of locality even at high noise, and somewhat sparse dependencies on the text conditioner (measured by splitting the prompt into individual words and computing the score delta as we ablate each word). For example, the region where the horse ends up being drawn has the strongest dependency on the word "horse" at most noise levels, similar to observations in Chefer et al. (2023); Hertz et al. (2022). The local structure is far from perfect, but notably so is the compositional generalization: it fails to draw a cat as requested in the prompt. Details in Appendix G and F.

Next, we move beyond pixel-space to study F-LCS structure within the learned feature-spaces of SDXL. Certain concepts, such as animals vs. art styles, almost certainly interact in pixel-space but might be disentangled in the network's internal representation. In fact, there is significant evidence suggesting concept disentanglement (according to various metrics) within the learned feature-spaces of large-scale diffusion models (Karras et al., 2019; Kotovenko et al., 2019; Gatys et al., 2016; Zhu et al., 2017), which may help to explain their compositional abilities. To connect directly with our theory, we measure our F-LCS disentanglement heuristic given by Lemma 2, and connect this with

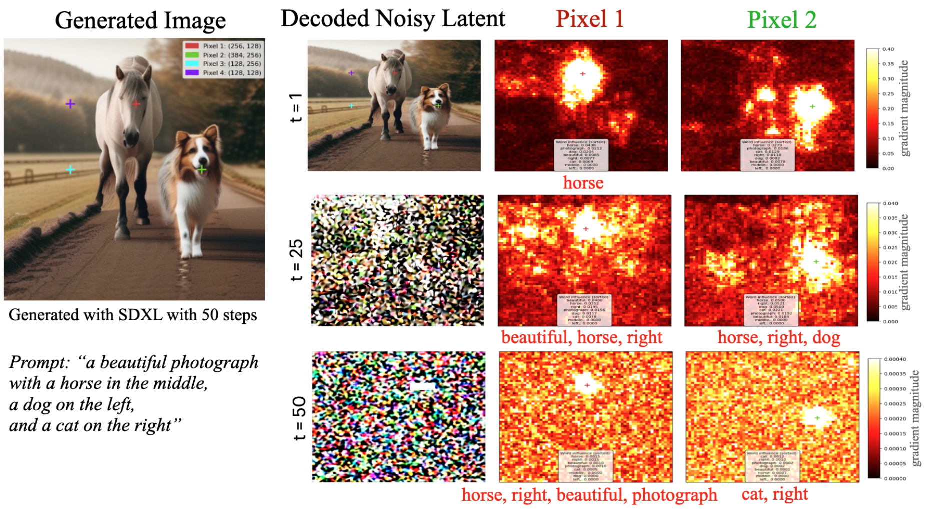

Figure 5: **Preliminary SDXL pixel-space locality study**. An SDXL generated image with the prompt "a beautiful photograph with a horse in the middle, a dog on the left, and a cat on the right." Heatmaps show pixel gradient magnitude at locations marked with $+$. The conditional gradient magnitude w.r.t. individual words is also evaluated at the indicated pixel, with dominant words (if any) shown in red. SDXL shows *some degree* of pixel-locality (particularly at low noise) and *some degree* of conditional-locality (more so at higher noise); consistent with apparently only *some degree* of compositional generalization (note the failure to draw a cat).

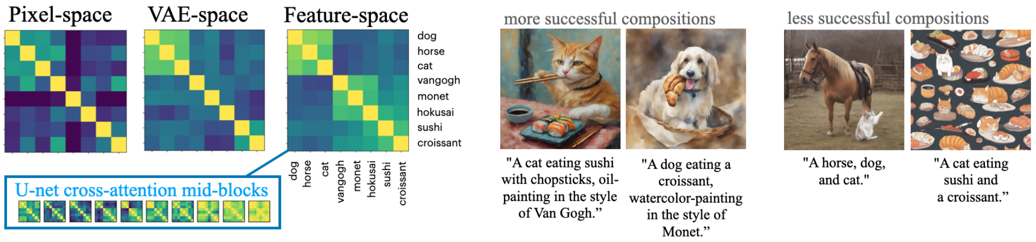

Figure 6: **Preliminary evidence for feature-space compositionality in SDXL.** (Left) F-LCS disentanglement between concepts (dog, horse, cat, van Gogh, Monet, Hokusai, sushi, croissant) via the Lemma 2 heuristic (low cosine similarity off-diagonal indicates L-FCS). *Pixel-space* lacks meaningful structure, *VAE-space* exhibits some structure, and a clearer structure emerges within a proposed compositional *Feature-space* comprised of U-net mid block activations (see Figure 16), showing higher intra-group similarity (e.g. dog, horse, cat) and lower inter-group similarity (e.g. cat, van Gogh). (Right) SDXL example generations. F-LCS-disentangled concepts in the proposed feature-space (e.g. cat+sushi+van Gogh) compose more successfully than highly-entangled concepts (e.g. horse+dog+cat).

compositional generalization. Specifically, we analyze U-net cross-attention layer activations via the Lemma 2 heuristic, and find evidence of F-LCS structure within several layers of the midblock: related concepts like dog, cat, and horse have higher similarity, while concepts like cat and vanGogh have lower similarity. We also show examples connecting this feature-space structure with successful and unsuccessful compositions. Details in Appendix G.

5 Discussion and Future Work

Whether local mechanisms can explain compositional generalization in real-world diffusion models remains an open question. The preliminary SDXL experiments in Section 4 are meant only to be suggestive, and much more work is needed to fully understand compositional generalization in modern text-to-image models. This setting presents several challenges. First, since we often don't know what was in the training set, it is unclear which prompts are actually OOD (however, see Appendix H for a small exploratory study of a model trained on a known dataset). Second, some kinds of compositional structure, such as style+content, exist only in feature-space, requiring more complex studies of the model's learned representation as in Figures 4 and 6, potentially relying on heuristics such as Lemma 2. Despite significant evidence for the existence of disentangled feature spaces (Chen et al., 2018; Kim & Mnih, 2018; Yang et al., 2023; Locatello et al., 2019; Zhang et al., 2023; Ilharco et al., 2022) and diffusion models' ability to learn them in some cases Karras et al. (2019); Kotovenko et al. (2019); Gatys et al. (2016); Zhu et al. (2017), disentanglement in diffusion representations is still not fully understood. Future work could also exploit our finding of local conditional scores as a mechanism to improve compositional generalization. Our causal intervention in Experiment 2L shows that in a simple setting, enforcing an explicitly local architecture can improve generalization, suggesting that similar architectural, training, or inference-based interventions might be able to improve real-world models. Several existing methods can be viewed as implicit "local interventions": for instance, layout-to-image methods that use explicit local constraints or biases (see Appendix A), or more generally, sparse attention architectures Child et al. (2019); Sun et al. (2022). Our theory helps to explain why these approaches are beneficial, and suggests more precise ways to target compositional generalization by specifically enforcing local conditional scores. When compositional structure exists only in feature-space, such interventions become more complex. The challenge becomes two-fold: we must first identify – or attempt to induce – feature-space transformations that reveal the compositional structure, and then apply local interventions within the learned feature space: perhaps via sparsity-inducing regularization (such as L1) or explicitly-sparse architectures. Larger-scale studies on complex, real-world datasets are essential to clarify the challenges and explore opportunities to improve compositional generalization via local interventions.

6 Conclusion

We proposed local conditional scores as a possible mechanism for compositional generalization. Theoretically, we proved an equivalence between conditional projective composition (a specific compositional structure) and local conditional scores (which capture both pixel- and conditional-locality); this theory extends to feature-space compositionality. Empirically, we verified that length generalization in location-conditioned CLEVR models corresponds with local conditional dependencies at high noise combined with pixel-locality at low noise. Then, we demonstrated through a causal intervention that enforcing a local architecture restores length generalization in a model that previously failed. We also offered evidence for feature-space compositionality in color-conditioned CLEVR, and preliminary evidence of compositional structure in both pixel- and feature-space in SDXL. Our results support local conditional scores as a potential mechanism of compositional generalization in conditional diffusion models, offering a lens to understand when and how generalization is achieved, and potential avenues to improve it.

References

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

Quentin Bertrand, Anne Gagneux, Mathurin Massias, and Rémi Emonet. On the closed-form of flow matching: Generalization does not arise from target stochasticity. *arXiv preprint arXiv:2506.03719*, 2025.

Arwen Bradley, Preetum Nakkiran, David Berthelot, James Thornton, and Joshua M Susskind. Mechanisms of projective composition of diffusion models. *arXiv preprint arXiv:2502.04549*, 2025.

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 (TOG)*, 42(4):1–10, 2023.

Ricky TQ Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating sources of disentanglement in variational autoencoders. *Advances in neural information processing systems*, 31, 2018.

Siyi Chen, Huijie Zhang, Minzhe Guo, Yifu Lu, Peng Wang, and Qing Qu. Exploring low-dimensional subspace in diffusion models for controllable image editing. *Advances in neural information processing systems*, 37:27340–27371, 2024.

Jiaxin Cheng, Xiao Liang, Xingjian Shi, Tong He, Tianjun Xiao, and Mu Li. Layoutdiffuse: Adapting foundational diffusion models for layout-to-image generation. *arXiv preprint arXiv:2302.08908*, 2023.

Jiaxin Cheng, Zixu Zhao, Tong He, Tianjun Xiao, Zheng Zhang, and Yicong Zhou. Rethinking the training and evaluation of rich-context layout-to-image generation. *Advances in Neural Information Processing Systems*, 37:62083–62107, 2024.

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019.

Omer Dahary, Or Patashnik, Kfir Aberman, and Daniel Cohen-Or. Be yourself: Bounded attention for multi-subject text-to-image generation. In *European Conference on Computer Vision*, pp. 432–448. Springer, 2024.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Yilun Du and Leslie Pack Kaelbling. Position: Compositional generative modeling: A single model is not all you need. In *Forty-first International Conference on Machine Learning*, 2024.

Yilun Du, Conor Durkan, Robin Strudel, Joshua B Tenenbaum, Sander Dieleman, Rob Fergus, Jascha Sohl-Dickstein, Arnaud Doucet, and Will Sussman Grathwohl. Reduce, reuse, recycle: Compositional generation with energy-based diffusion models and mcmc. In *International conference on machine learning*, pp. 8489–8510. PMLR, 2023.

Alessandro Favero, Antonio Sclocchi, Francesco Cagnetta, Pascal Frossard, and Matthieu Wyart. How compositional generalization and creativity improve as diffusion models are trained. *arXiv preprint arXiv:2502.12089*, 2025.

Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2414–2423, 2016.

Jiatao Gu, Shuangfei Zhai, Yizhe Zhang, Josh Susskind, and Navdeep Jaitly. Matryoshka diffusion models, 2023a. URL https://arxiv.org/abs/2310.15111.

Xiangming Gu, Chao Du, Tianyu Pang, Chongxuan Li, Min Lin, and Ye Wang. On memorization in diffusion models. *arXiv preprint arXiv:2310.02664*, 2023b.

Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *International conference on learning representations*, 2017.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. *arXiv preprint arXiv:2212.04089*, 2022.

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*, pp. 2901–2910, 2017.

Zahra Kadkhodaie, Florentin Guth, Eero P Simoncelli, and Stéphane Mallat. Generalization in diffusion models arises from geometry-adaptive harmonic representations. *arXiv preprint arXiv:2310.02557*, 2023.

Mason Kamb and Surya Ganguli. An analytic theory of creativity in convolutional diffusion models. *arXiv preprint arXiv:2412.20292*, 2024.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4401–4410, 2019.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022.

Tero Karras, Miika Aittala, Jaakko Lehtinen, Janne Hellsten, Timo Aila, and Samuli Laine. Analyzing and improving the training dynamics of diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24174–24184, 2024.

Hyunjik Kim and Andriy Mnih. Disentangling by factorising. In *International conference on machine learning*, pp. 2649–2658. PMLR, 2018.

Dmytro Kotovenko, Artsiom Sanakoyeu, Sabine Lang, and Bjorn Ommer. Content and style disentanglement for artistic style transfer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4422–4431, 2019.

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*, pp. 22511–22521, 2023.

Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B Tenenbaum. Compositional visual generation with composable diffusion models. In *European Conference on Computer Vision*, pp. 423–439. Springer, 2022.

Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *international conference on machine learning*, pp. 4114–4124. PMLR, 2019.

Artem Lukoianov, Chenyang Yuan, Justin Solomon, and Vincent Sitzmann. Locality in image diffusion models emerges from data statistics. *arXiv preprint arXiv:2509.09672*, 2025.

Matthew Niedoba, Berend Zwartsenberg, Kevin Murphy, and Frank Wood. Towards a mechanistic explanation of diffusion model generalization. *arXiv preprint arXiv:2411.19339*, 2024.

Maya Okawa, Ekdeep S Lubana, Robert Dick, and Hidenori Tanaka. Compositional abilities emerge multiplicatively: Exploring diffusion models on a synthetic task. *Advances in Neural Information Processing Systems*, 36, 2024.

Core Francisco Park, Maya Okawa, Andrew Lee, Ekdeep S Lubana, and Hidenori Tanaka. Emergence of hidden capabilities: Exploring learning dynamics in concept space. *Advances in Neural Information Processing Systems*, 37:84698–84729, 2024.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022.

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*, pp. 10684–10695, 2022.

Antonio Sclocchi, Alessandro Favero, and Matthieu Wyart. A phase transition in diffusion models reveals the hierarchical nature of data. *Proceedings of the National Academy of Sciences*, 122(1): e2408799121, 2025.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pp. 2256–2265. pmlr, 2015.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. URL `https://arxiv.org/pdf/2011.13456.pdf`.

Yutao Sun, Li Dong, Barun Patra, Shuming Ma, Shaohan Huang, Alon Benhaim, Vishrav Chaudhary, Xia Song, and Furu Wei. A length-extrapolatable transformer. 2022.

Peng Wang, Huijie Zhang, Zekai Zhang, Siyi Chen, Yi Ma, and Qing Qu. Diffusion models learn low-dimensional distributions via subspace clustering. *arXiv preprint arXiv:2409.02426*, 2024.

Nicholas Watters, Loic Matthey, Christopher P Burgess, and Alexander Lerchner. Spatial broadcast decoder: A simple architecture for learning disentangled representations in vaes. *arXiv preprint arXiv:1901.07017*, 2019.

taemin6697 wooyeolbaek, mbaek01. attention-map-diffusers. `https://github.com/wooyeolbaek/attention-map-diffusers`, 2025.

Jinheng Xie, Yuexiang Li, Yawen Huang, Haozhe Liu, Wentian Zhang, Yefeng Zheng, and Mike Zheng Shou. Boxdiff: Text-to-image synthesis with training-free box-constrained diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7452–7461, 2023.

Han Xue, Zhiwu Huang, Qianru Sun, Li Song, and Wenjun Zhang. Freestyle layout-to-image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14256–14266, 2023.

Zhutian Yang, Jiayuan Mao, Yilun Du, Jiajun Wu, Joshua B Tenenbaum, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. Compositional diffusion-based continuous constraint solvers. *arXiv preprint arXiv:2309.00966*, 2023.

Jinghan Zhang, Junteng Liu, Junxian He, et al. Composing parameter-efficient modules with arithmetic operation. *Advances in Neural Information Processing Systems*, 36:12589–12610, 2023.

Guangcong Zheng, Xianpan Zhou, Xuewei Li, Zhongang Qi, Ying Shan, and Xi Li. Layoutdiffusion: Controllable diffusion model for layout-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22490–22499, 2023.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 2223–2232, 2017.

A Additional Related Work

Locality and generalization in diffusion models Kamb & Ganguli (2024); Niedoba et al. (2024) argue that "creativity" in unconditional and class-conditional models arises from a bias toward learning local denoisers, while Lukoianov et al. (2025) challenge some of the conclusions of Kamb & Ganguli (2024) by arguing that locality arises from statistical properties of the data rather than network inductive bias. Our work builds on the idea of locality by showing that it can also be a mechanism for compositional generalization in conditional models, which none of the previous works addressed. We are agnostic about whether locality arises from data statistics or inductive bias in practice (likely both are at play); our main insight is that compositional generalization can be achieved when the network learns a local – or equivalently, compositional – structure by any means. (Our Experiment 2L suggests that local architectural interventions can promote learning the underlying the compositional structure of the data, but this structure can also be learned "naturally" as in Experiment 1.) Note also that Kamb & Ganguli (2024) propose equivariance while Niedoba et al. (2024); Lukoianov et al. (2025) omit or argue against it; in our theory and experiments equivariance is not necessary.

Generalization of diffusion models Kadkhodaie et al. (2023) propose shrinkage in a geometry-adaptive harmonic bias as a mechanism for generalization; in the framework of our theory, this can be thought of as a bias toward sparse dependencies in a particular feature-space. Gu et al. (2023b) make a experimentally study of potential causes generalization in unconditional and class-conditional settings based on the characteristics of the dataset and choices for training and model. Bertrand et al. (2025) give empirical evidence that for flow models in high-dimensions, generalization arises primarily from network inductive biases rather than noise in the flow-matching loss.

Learning Dynamics of Composition An interesting line of work focuses on the learning dynamics of models trained on compositional data. Okawa et al. (2024); Park et al. (2024) demonstrate sudden emergence of compositional generalization in controlled synthetic settings where they test novel combinations of attributes of a single object (e.g. blue square, red triangle → blue triangle?). Sclocchi et al. (2025); Favero et al. (2025) study learning dynamics in hierarchical models, showing that higher-level features take longer to learn.

Explicit composition of multiple diffusion models Du & Kaelbling (2024); Liu et al. (2022); Bradley et al. (2025) study *explicit* compositions of multiple diffusion models via linear score combination, demonstrating length-generalization in CLEVR in some cases. Bradley et al. (2025) proposed Projective Composition as a definition of "correct" compositions of multiple models; here we use it to precisely characterize compositionality in a single conditional model.

Layout-to-Image Diffusion Models Our theory may help to explain the success of layout-to-image methods that use architectural locality priors (Li et al., 2023; Zheng et al., 2023; Cheng et al., 2023; 2024) or inference-time locality constraints (Dahary et al., 2024; Xue et al., 2023; Xie et al., 2023) to improve multi-object generation. Although these works primarily report improved grounding and controllability rather than explicit OOD composition, their interventions can be viewed as approximate causal tests, where increased locality leads to improved multi-object behavior, consistent with our theory and similar to the causal intervention Exp. 2L.

Feature-space disentanglement There is a large body of work towards designing disentanglement-metrics appropriate for "real-world" distributions (e.g. disentanglement metrics introduced by BetaVAE Higgins et al. (2017), FactorVAE Kim & Mnih (2018); MIG in Chen et al. (2018), etc.). Many works have shown evidence of disentanglement over a variety of datasets such as CelebA Karras et al. (2019); Chen et al. (2018); Kim & Mnih (2018), Shapes3D Locatello et al. (2019), and dSprites Watters et al. (2019); Chen et al. (2018); also Kotovenko et al. (2019) explores disentanglement between style and content for style transfer. Other works have shown that diffusion models have some ability to learn disentangled feature spaces Karras et al. (2019); Kotovenko et al. (2019); Gatys et al. (2016); Zhu et al. (2017). Nevertheless disentanglement in diffusion representations is still not fully understood.

B Proof of Lemma 1

Proof. (Lemma 1) Let p be the CPC given by

$$p(x|c_{\mathcal{J}}) := p(x_{M_{\mathcal{J}}^{\complement}}|\emptyset) \prod_{j \in \mathcal{J}} p(x_{M_j}|c_j), \quad \forall \mathcal{J} \in \mathcal{J}_{\text{all}}$$

and let s be the LCS given by

$$\begin{aligned} s^t[x|c_{\mathcal{J}}](i) &:= \nabla \log p^t[x_{N_i^t}|c_{L_i^t(\mathcal{J})}](i), \quad \forall i, \forall t \\ L_i^t(\mathcal{J}) &= \begin{cases} \{j\} \cap \mathcal{J}, & \text{if } i \in M_j \\ \emptyset, & \text{else} \end{cases} \\ N_i^t &= \begin{cases} M_j, & \text{if } i \in M_j \\ M_b, & \text{else,} \end{cases} \qquad \text{where } M_b := M^{\complement}_{\mathcal{J}_{\text{all}}}. \end{aligned}$$

We want to show that s is exactly the score of p:

$$s^t(x|c_{\mathcal{J}}) = \nabla \log p^t(x|c_{\mathcal{J}}), \quad \forall \mathcal{J}.$$

To see this, we first analyze p^t at each pixel i. We begin by noting that if p has a pixel-space PC structure at time 0 then it has an identical PC structure at all times t (Bradley et al., 2025) (because adding isotropic Gaussian noise preserves the independence between subsets).

$$\begin{aligned} \nabla \log p^t(x|c_{\mathcal{J}}) &:= \nabla \log p^t(x_{M^{\complement}_{\mathcal{J}}}|\emptyset) + \sum_{j \in \mathcal{J}} \nabla \log p^t(x_{M_j}|c_j), \quad \forall \mathcal{J} \\ \nabla \log p^t(x_{M_j}|c_j)(i) = 0, &\quad \forall i \notin M_j, \quad \text{since } p^t(x_{M_j}) \text{ does not depend on } x_i \\ \implies j \in \mathcal{J}, \quad i \in M_j &\implies \nabla \log p^t(x|c_{\mathcal{J}})(i) = \nabla \log p^t(x_{M_j}|c_j)(i) \\ j \notin \mathcal{J}, \quad i \in M_j &\implies \nabla \log p^t(x|c_{\mathcal{J}})(i) = \nabla \log p^t(x_{M_j}|\emptyset)(i) \\ i \in M_b &\implies \nabla \log p^t(x|c_{\mathcal{J}})(i) = \nabla \log p^t_b(x_{M_b}|\emptyset)(i). \end{aligned}$$

Next we analyze s at each pixel i:

$$\begin{aligned} s^t[x|c_{\mathcal{J}}](i) &:= \nabla \log p^t[x_{N_i^t}|c_{L_i^t(\mathcal{J})}](i) \\ j \in \mathcal{J}, \quad i \in M_j \implies L_i^t(\mathcal{J}) = \{j\}, \quad N_i^t = M_j &\implies s^t[x|c_{\mathcal{J}}](i) = \nabla \log p^t(x_{M_j}|c_j)(i) \\ j \notin \mathcal{J}, \quad i \in M_j \implies L_i^t(\mathcal{J}) = \emptyset, \quad N_i^t = M_j &\implies s^t[x|c_{\mathcal{J}}](i) = \nabla \log p^t(x_{M_j}|\emptyset)(i) \\ i \in M_b \implies L_i^t = \emptyset, \quad M_b \subseteq N_i^t &\implies s^t(x|c_{\mathcal{J}})(i) = \nabla \log p^t_b[x_{M_b}|\emptyset](i). \end{aligned}$$

Comparing $s^t(x|c_{\mathcal{J}})(i)$ and $\nabla \log p^t(x|c_{\mathcal{J}})(i)$ in each of the three cases, we see that $s^t[x|c](i) = \nabla \log p^t(x|c)(i)$ for all pixels i, hence

$$s^t(x|c_{\mathcal{J}}) = \nabla \log p^t(x|c_{\mathcal{J}}), \quad \forall \mathcal{J}.$$

□

C Relaxation of Lemma 1

In this section we provide a collection of lemmas showing that the score of an approximately CPC distribution is approximately an LCS, and that this approximation becomes more accurate as noise increases. We state all lemmas first and then provide the proofs at the end.

We begin by defining a notion of *approximate* conditional projective composition.

Definition 3 (Approximate CPC). *We say that a conditional distribution $p^t(x|c)$ is approximately-CPC with errors $\{\varepsilon_j, \tilde{\varepsilon}_j, \varepsilon_b\}$ if*

$$\sup_{x_{M_j^{\complement}}} D_{\text{KL}}[p(x_{M_j}|c_{\mathcal{J}}, x_{M_j^{\complement}})||p(x_{M_j}|c_j)] \le \varepsilon_j, \quad \forall \mathcal{J}, \quad \forall j \in \mathcal{J} \tag{3}$$

$$\sup_{x_{M_j^{\complement}}} D_{\text{KL}}[p(x_{M_j}|c_{\mathcal{J}}, x_{M_j^{\complement}})||p(x_{M_j}|\emptyset)] \le \tilde{\varepsilon}_j, \quad \forall \mathcal{J}, \quad \forall j \notin \mathcal{J} \tag{4}$$

$$\sup_{x_{M_b^{\complement}}} D_{\text{KL}}[p(x_{M_b}|c_{\mathcal{J}}, x_{M_b^{\complement}})||p(x_{M_b})] \le \varepsilon_b, \quad \forall \mathcal{J}. \tag{5}$$

The following lemma is relaxation of Lemma 1. It shows that the score of an *approximately-CPC* distribution is *approximately* an LCS.

Lemma 3 (LCS approximates score of approximate-CPC)**.** *Let $p^t(x|c)$ be approximately-CPC with errors $\{\varepsilon_j, \tilde{\varepsilon}_j, \varepsilon_b\}$ per Definition 3. Define a local conditional score s as in Lemma 1, and let $\widehat{p}$ be the induced distribution s.t. $s^t(x|c_{\mathcal{J}}) = \nabla \log \widehat{p}^t(x|c_{\mathcal{J}})$. Then*

$$D_{\mathrm{KL}}(p(\cdot|c_{\mathcal{J}})||\widehat{p}(\cdot|c_{\mathcal{J}})) \le \sum_{j\in\mathcal{J}} \varepsilon_j + \sum_{j\notin\mathcal{J}} \tilde{\varepsilon}_j + \varepsilon_b.$$

Further, we can show that the approximation errors in Definition 3 decrease as noise is added: intuitively, an approximately-compositional distribution gets *more compositional* as noise is added.

Lemma 4. *Suppose that the supremum of $\sup_y[KL(N_t[p](x)||N_t[p](x|y))]$ is attained for all t. Then*

$$\sup_y[KL(N_t[p](x)||N_t[p](x|y))]$$

is decreasing in t.

The proof of Lemma 4 essentially follows from the fact that adding Gaussian noise decreases the KL divergence between distributions:

Claim 1 (Standard; KL divergence decreases with noise)**.**

$$\frac{\partial}{\partial t} D_{\mathrm{KL}}(N_t[q]||N_t[r]) = -t\left[\left(\nabla \log \frac{N_t[q]}{N_t[r]}\right)^2\right] < 0, \quad \textit{if } q \neq r.$$

This is a standard fact but a proof is offered for the reader's convenience at the end of this section. Note that the Data Processing Inequality immediately implies that the KL divergence is non-increasing, but the claim is that it is actually decreasing.

Combining Lemma 3 and Lemma 4, we see that at high noise levels, distributions become *more compositional*, and thus better-approximated by local-conditional-scores. We now provide the proofs.

Proof. (Lemma 3)

Define the "ideal" projective composition for any $\mathcal{J}$ by:

$$\mathcal{C}^{\star}_{\mathcal{J}}[p](x) := p(x_{M^{\complement}_{\mathcal{J}}}|\emptyset) \prod_{j\in\mathcal{J}} p(x_{M_j}|c_j).$$

By Lemma 1, we have that s is exact for $\nabla \log \mathcal{C}^{\star}[p]$, and so $\widehat{p}(\cdot|c_{\mathcal{J}}) = cC^{\star}_{\mathcal{J}}[p]$.

Thus we need to show that

$$D_{\mathrm{KL}}(p(\cdot|c_{\mathcal{J}})||\mathcal{C}^{\star}_{\mathcal{J}}[p]) \le \sum_{j\in\mathcal{J}} \varepsilon_j + \sum_{j\notin\mathcal{J}} \tilde{\varepsilon}_j + \varepsilon_b.$$

This is essentially a bound on a mean field approximation. First, note that for any c, we can rewrite $p(x|c)$ using the chain rule as

$$p(x|c) = p(x_{M_b}|c_{\mathcal{J}}) \prod_{j\in\mathcal{J}_{\text{all}}} p(x_{M_j}|c, x_{M_b}, x_{M_1}, \ldots, x_{M_{j-1}})$$

Then calculate

$$
\begin{aligned}
D_{\mathrm{KL}}(p(\cdot|c_{\mathcal{J}})||\mathcal{C}^\star_{\mathcal{J}}[p]) &\equiv \mathbb{E}_{p(x|c_{\mathcal{J}})}\left[\log \frac{p(x|c_{\mathcal{J}})}{\mathcal{C}^\star[\vec{p}](x)}\right] \\
&= \mathbb{E}_{p(x|c_{\mathcal{J}})}\left[\log \frac{p(x_{M_b}|c_{\mathcal{J}})\prod_j p(x_{M_j}|c_{\mathcal{J}}, x_{M_b}, x_{M_1}, \dots, x_{M_{j-1}})}{p(x_{M_b})\prod_{j\notin\mathcal{J}} p(x_{M_j}|\emptyset)\prod_{j\in\mathcal{J}} p(x_{M_j}|c_j)}\right] \\
&= \sum_{j\in\mathcal{J}} \mathbb{E}_{p(x|c_{\mathcal{J}})}\left[\log \frac{p(x_{M_j}|c_{\mathcal{J}}, x_{M_b}, x_{M_1}, \dots, x_{M_{j-1}})}{p(x_{M_j}|c_j)}\right] \\
&\quad + \sum_{j\notin\mathcal{J}} \mathbb{E}_{p(x|c_{\mathcal{J}})}\left[\log \frac{p(x_{M_j}|c_{\mathcal{J}}, x_{M_b}, x_{M_1}, \dots, x_{M_{j-1}})}{p(x_{M_j}|\emptyset)}\right] \\
&\quad + \mathbb{E}_p\left[\log \frac{p(x_{M_b}|c_{\mathcal{J}})}{p(x_{M_b})}\right] \\
&= \sum_{j\in\mathcal{J}} \mathbb{E}_{p(x_{M_b}, x_{M_1}, \dots, x_{M_{j-1}}|c_{\mathcal{J}})}\left[D_{\mathrm{KL}}[p(x_{M_j}|c_{\mathcal{J}}, x_{M_b}, x_{M_1}, \dots, x_{M_{j-1}})||p(x_{M_j}|c_j))\right] \\
&\quad + \sum_{j\notin\mathcal{J}} \mathbb{E}_{p(x_{M_b}, x_{M_1}, \dots, x_{M_{j-1}}|c_{\mathcal{J}})}\left[D_{\mathrm{KL}}[p(x_{M_j}|c_{\mathcal{J}}, x_{M_b}, x_{M_1}, \dots, x_{M_{j-1}})||p(x_{M_j}|\emptyset))\right] \\
&\quad + \mathbb{E}_{p(x_{M_b^\complement}|c_{\mathcal{J}})}\left[D_{\mathrm{KL}}[p(x_{M_b}|c_{\mathcal{J}})||p(x_{M_b}))\right] \\
&\leq \sum_{j\in\mathcal{J}} \sup_{x_{M_j^\complement}} D_{\mathrm{KL}}[p(x_{M_j}|c_{\mathcal{J}}, x_{M_j^\complement})||p(x_{M_j}|c_j)] \\
&\quad + \sum_{j\notin\mathcal{J}} \sup_{x_{M_j^\complement}} D_{\mathrm{KL}}[p(x_{M_j}|c_{\mathcal{J}}, x_{M_j^\complement})||p(x_{M_j}|\emptyset)] \\
&\quad + \sup_{x_{M_b^\complement}} \quad D_{\mathrm{KL}}[p(x_{M_b}|c_{\mathcal{J}}, x_{M_b^\complement})||p(x_{M_b})] \\
&\leq \sum_{j\in\mathcal{J}} \varepsilon_j + \sum_{j\notin\mathcal{J}} \tilde{\varepsilon}_j + \varepsilon_b
\end{aligned}
$$

□

Proof. (Lemma 4)

We want to show that

$$t_2 \geq t_1 \implies \sup_y[KL(N_{t_2}[p](x)||N_{t_2}[p](x|y))] < \sup_y[KL(N_{t_1}[p](x)||N_{t_1}[p](x|y))]$$

Let $t_2 \geq t_1$. By assumption, the supremum of $\sup_y[KL(N_{t_2}[p](x)||N_{t_2}[p](x|y))]$ is attained, so let y_2^* be a value of y that achieves the supremum for t_2. Then

$$
\begin{aligned}
y_2^\star &:= \arg\max_y[KL(N_{t_2}[p](x)||N_{t_2}[p](x|y))] \\
\sup_y[KL(N_{t_1}[p](x)||N_{t_1}[p](x|y))] &:= KL(N_{t_2}[p](x)||N_{t_2}[p](x|y_2^\star)) \\
&\leq KL(N_{t_1}[p](x)||N_{t_1}[p](x|y_2^\star)), \quad \text{by Claim 1} \\
&\leq \sup_y[KL(N_{t_1}[p](x)||N_{t_1}[p](x|y))]
\end{aligned}
$$

□

Proof. (Claim 1) We want to show that

$$\frac{\partial}{\partial t} D_{\mathrm{KL}}(N_t[q]||N_t[r]) = -t\left[\left(\nabla \log \frac{N_t[q]}{N_t[r]}\right)^2\right] < 0, \quad \text{if } q \neq r.$$

This is a standard result but we provide a proof for the reader's convenience.

$$
\begin{aligned}
D_{\mathrm{KL}}(q||r) := \mathbb{E}_q[\log \frac{q}{r}] &= \int q(x) \log \frac{q(x)}{r(x)} dx \\
N_t[p](x) &:= \int p(y) \mathcal{N}(x; y, t^2) dy \\
\mathbb{E}_{N_t[q]}[\log \frac{N_t[q]}{N_t[r]}] &= \int N_t[q](x) \log \frac{N_t[q](x)}{N_t[r](x)} dx \\
\frac{\partial}{\partial t}\mathbb{E}_{N_t[q]}[\log \frac{N_t[q]}{N_t[r]}] &= \frac{\partial}{\partial t} \int N_t[q](x) \log \frac{N_t[q](x)}{N_t[r](x)} dx \\
&= \int \frac{\partial}{\partial t} N_t[q](x) \cdot \log \frac{N_t[q](x)}{N_t[r](x)} dx + \int N_t[q](x) \cdot \frac{\partial}{\partial t} \log \frac{N_t[q](x)}{N_t[r](x)} dx \\
&\equiv I_1 + I_2
\end{aligned}
$$

To work on integrals I_1, I_2, we use the following fact: $\frac{\partial}{\partial t} N_t[p](x) = t\nabla^2 N_t[p](x)$ (Equation (6)).

$$
\begin{aligned}
I1 &\equiv \int \frac{\partial}{\partial t} N_t[q](x) \cdot \log \frac{N_t[q](x)}{N_t[r](x)} dx \\
&= \int t\nabla^2 N_t[q](x) \cdot \log \frac{N_t[q](x)}{N_t[r](x)} dx, \quad \text{using } \frac{\partial}{\partial t} N_t[p] = t\nabla^2 N_t[p] \text{ as shown below} \\
&= -t \int \nabla N_t[q](x) \cdot \nabla \log \frac{N_t[q](x)}{N_t[r](x)} dx, \quad \text{integration by parts} \\
&= -t \int \nabla N_t[q](x) \cdot \left(\nabla \log N_t[q](x) - \nabla \log N_t[r](x)\right) dx, \\
&= -t \int \nabla N_t[q] \cdot \left(\frac{\nabla N_t[q]}{N_t[q]} - \frac{\nabla N_t[r]}{N_t[r]} \right) dx \\
&= t \int -\frac{\nabla N_t[q]^2}{N_t[q]} + \frac{\nabla N_t[r] \nabla N_t[q]}{N_t[r]} dx
\end{aligned}
$$

$$
\begin{aligned}
I_2 &\equiv \int N_t[q](x) \cdot \frac{\partial}{\partial t} \log \frac{N_t[q](x)}{N_t[r](x)} dx \\
\frac{\partial}{\partial t} \log \frac{N_t[q](x)}{N_t[r](x)} &= \frac{\partial}{\partial t} \log N_t[q](x) - \frac{\partial}{\partial t} \log N_t[r](x) \\
&= \frac{1}{N_t[q](x)} \frac{\partial}{\partial t} N_t[q](x) - \frac{1}{N_t[r](x)} \frac{\partial}{\partial t} N_t[r](x) \\
&= \frac{t}{N_t[q](x)} \nabla^2 N_t[q](x) - \frac{t}{N_t[r](x)} \nabla^2 N_t[r](x) \\
\implies I_2 &= t \int N_t[q](x) \cdot \left(\frac{\nabla^2 N_t[q](x)}{N_t[q](x)} - \frac{\nabla^2 N_t[r](x)}{N_t[r](x)} \right) dx \\
&= -t \int \frac{N_t[q](x)}{N_t[r](x)} \nabla^2 N_t[r] dx, \quad \text{since } \int \nabla^2 N_t[q] dx = \nabla N_t[q]|_{-\infty}^{\infty} = 0 \\
&= t \int \nabla \left(\frac{N_t[q](x)}{N_t[r](x)} \right) \nabla N_t[r] dx, \quad \text{integration by parts} \\
&= t \int \frac{\nabla N_t[r] \nabla N_t[q]}{N_t[r]} - \frac{\nabla N_t[r]^2 N_t[q]}{N_t[r]^2} dx
\end{aligned}
$$

Therefore

$$\begin{aligned}
I_1 + I_2 &= t \int -\frac{\nabla N_t[q]^2}{N_t[q]} + \frac{\nabla N_t[r] \nabla N_t[q]}{N_t[r]} dx + t \int \frac{\nabla N_t[r] \nabla N_t[q]}{N_t[r]} - \frac{\nabla N_t[r]^2 N_t[q]}{N_t[r]^2} dx \\
&= -t \int \frac{\nabla N_t[q]^2}{N_t[q]} - 2\frac{\nabla N_t[r] \nabla N_t[q]}{N_t[r]} + \frac{\nabla N_t[r]^2 N_t[q]}{N_t[r]^2} dx \\
&= -t \int N_t[q] \left(\frac{\nabla N_t[q]}{N_t[q]} - \frac{\nabla N_t[r]}{N_t[r]} \right)^2 dx \\
&= -t \left[\left(\nabla \log \frac{N_t[q]}{N_t[r]} \right)^2 \right]
\end{aligned}$$

This concludes the proof. Note that the fact used earlier can be shown as follows:

$$\text{Claim:} \quad \frac{\partial}{\partial t} N_t[p](x) = t \nabla^2 N_t[p](x) \tag{6}$$

To see this:

$$\begin{aligned}
N_t[p](x) &:= \int p(y) \phi(y) dy, \quad \phi(y; x, t) := \frac{1}{\sqrt{2\pi t^2}} e^{-\frac{(x-y)^2}{2t^2}} \\
\frac{\partial}{\partial t} \phi(y; x, t) &= \frac{1}{\sqrt{2\pi}} \left[-\frac{1}{t^2} + \frac{(x-y)^2}{t^4} \right] e^{-\frac{(x-y)^2}{2t^2}} = \left[\frac{(x-y)^2}{t^3} - \frac{1}{t} \right] \phi(y; x, t) \\
\frac{\partial^2}{\partial x^2} \phi(y; x, t) &= \frac{1}{\sqrt{2\pi t^2}} \left[\frac{(x-y)^2 - t^2}{t^4} \right] e^{-\frac{(x-y)^2}{2t^2}} = \frac{1}{t} \left[\frac{(x-y)^2}{t^3} - \frac{1}{t} \right] \phi(y; x, t) \\
\implies \frac{\partial}{\partial t} N_t[p](x) &= \int p(y) \frac{\partial}{\partial t} \phi(y; x, t) dy = \int p(y) \left[\frac{(x-y)^2}{t^3} - \frac{1}{t} \right] \phi(y; x, t) dy \\
\nabla^2 N_t[p](x) &= \int p(y) \frac{\partial^2}{\partial x^2} \phi(y; x, t) dy = \int p(y) \frac{1}{t} \left[\frac{(x-y)^2}{t^3} - \frac{1}{t} \right] \phi(y; x, t) dy \\
\implies t \nabla^2 N_t[p](x) &= \frac{\partial}{\partial t} N_t[p](x)
\end{aligned}$$

□

D FEATURE-SPACE THEORY

In this section we discuss the relationship between CPC and LCS in feature space. Inspired by the feature-space adaptation of PC in Bradley et al. (2025), we define feature-space conditional projective composition as follows:

Definition 4 (Feature-space Conditional Projective Composition (F-CPC)). *We say that $p(x|c)$ is a F-CPC under an invertible transform $\mathcal{A} : \mathbb{R}^n \to \mathbb{R}^n$ (mapping pixel-space to feature-space), if $\mathcal{A} \sharp p$ (where $\sharp$ denotes the pushforward) is a CPC according to Definition 2, that is: there exist disjoint sets M_j for all conditions $j \in \mathcal{J}_{all}$ such that, for any set of conditions $\mathcal{J} \in \mathcal{J}_{all}$,*

$$(\mathcal{A} \sharp p)(z|c_{\mathcal{J}}) := (\mathcal{A} \sharp p)(z_{M^{\complement}_{\mathcal{J}}}|\emptyset) \prod_{j \in \mathcal{J}} (\mathcal{A} \sharp p)(z_{M_j}|c_j), \quad \textit{where } z := \mathcal{A}(x). \tag{7}$$

That is, p is an F-CPC if it has CPC compositional structure under an appropriate feature-space mapping. In order to exploit this sparse dependency structure, the model now needs to learn the associated feature-space transform and its inverse, in addition to the local subsets N_i, L_i. For F-CPC distributions, Corollary 1 follows directly from Lemma 1. We formally restate and prove Corollary 1:

Corollary 1 (LCS is exact for CPC in feature-space; formal)**.** *Suppose that $p(x|c)$ is an F-CPC (Definition 4) under an invertible transform $\mathcal{A} : \mathbb{R}^n \to \mathbb{R}^n$, with subsets $\{M_j : j \in \mathcal{J}_{all}\}$. Letting $z := \mathcal{A}(x)$ consider the specific local-conditional score s given by*

$$s^t_{\mathcal{A}}[z|c_{\mathcal{J}}](i) := \nabla_z \log(\mathcal{A} \sharp p)^t [z_{N^t_i} | c_{L^t_i(\mathcal{J})}](i),$$

where N_i, L_i are defined as in Lemma 1 w.r.t. the subsets $\{M_j\}$. Then $s^t_{\mathcal{A}}[z|c_{\mathcal{J}}]$ is exact for the score of $\mathcal{A} \sharp p(z|c_{\mathcal{J}})$ w.r.t. z:

$$s^t_{\mathcal{A}}(z|c_{\mathcal{J}}) = \nabla_z \log(\mathcal{A} \sharp p)^t (z|c_{\mathcal{J}}), \quad \forall \mathcal{J}.$$

Proof. Apply Lemma 1 to $\mathcal{A}\sharp p(z|c)$. □

Sampling with this local score is more complex than it looks, however! As discussed in Bradley et al. (2025), since the noising operator does not commute with $\mathcal{A}$, that is, $(\mathcal{A}\sharp p)^t \neq \mathcal{A}\sharp(p^t)$, sampling from p using $s^t_{\mathcal{A}}$ would actually requires the process

$$N_t[p] \xrightarrow{\mathcal{A}^t} N_t[\mathcal{A}\sharp p_i] \xrightarrow{s^t_{\mathcal{A}}} N_{t-1}[\mathcal{A}\sharp p] \xrightarrow{(\mathcal{A}^{t-1})^{-1}} N_{t-1}[p] \to \ldots \to p,$$

where N_t denotes the noising operator, i.e. $N_t[p] := p^t$, and $\mathcal{A}^t$ corrects for the non-commutativity:

$$\mathcal{A}^t\sharp N_t[p] := N_t[\mathcal{A}\sharp p].$$

Therefore, it is not enough to learn a single transform $\mathcal{A}$ and its inverse $\mathcal{A}^{-1}$ – the network actually needs to learn a time-dependent transform/inverse pair $\mathcal{A}^t$, $\mathcal{A}^{-1}$ accounting for the interaction between $\mathcal{A}$ and the noising process at each time t, which actually depends on the (unknown) distribution p. How feasible this is is currently unclear.

However, there are a few special-case worth noting. First, if $\mathcal{A}$ is an orthogonal transform then we have $\mathcal{A}^t = \mathcal{A}$ for all t. Second, we can show that as the noise level increases, the non-commutativity becomes less severe:

Claim 2.

$$\frac{\partial}{\partial t} D_{\mathrm{KL}}(N_t[\mathcal{A}\sharp p]||\mathcal{A}\sharp N_t[p]) < 0.$$

Thus, if the composition structure is most important for resolving global structure at high noise levels, it may be enough to learn the single transform $\mathcal{A}$ in order to approximately exploit the compositional structure.

Proof. (Claim 2) We want to show that

$$\frac{\partial}{\partial t} D_{\mathrm{KL}}(N_t[\mathcal{A}\sharp p]||\mathcal{A}\sharp N_t[p]) < 0.$$

$$\begin{aligned}
\frac{\partial}{\partial t} D_{\mathrm{KL}}(N_t[\mathcal{A}\sharp p]||\mathcal{A}\sharp N_t[p]) &= \frac{\partial}{\partial t}\int N_t[\mathcal{A}\sharp p] \log \frac{N_t[\mathcal{A}\sharp p]}{\mathcal{A}\sharp N_t[p])} \\
&= \int \frac{\partial}{\partial t} N_t[\mathcal{A}\sharp p] \cdot \log \frac{N_t[\mathcal{A}\sharp p]}{\mathcal{A}\sharp N_t[p])} + \int N_t[\mathcal{A}\sharp p] \cdot \frac{\partial}{\partial t} \log \frac{N_t[\mathcal{A}\sharp p]}{\mathcal{A}\sharp N_t[p])} \\
&:= I_1 + I_2
\end{aligned}$$

$$\begin{aligned}
I_1 &:= \int \frac{\partial}{\partial t} N_t[\mathcal{A}\sharp p] \cdot \log \frac{N_t[\mathcal{A}\sharp p]}{\mathcal{A}\sharp N_t[p])} \\
&= t\int \nabla^2 N_t[\mathcal{A}\sharp p] \cdot \log \frac{N_t[\mathcal{A}\sharp p]}{\mathcal{A}\sharp N_t[p])} \\
&= -t\int \nabla N_t[\mathcal{A}\sharp p] \cdot \nabla \log \frac{N_t[\mathcal{A}\sharp p]}{\mathcal{A}\sharp N_t[p])} \\
&= t\int -\frac{\nabla N_t[\mathcal{A}\sharp p]^2}{N_t[\mathcal{A}\sharp p]} + \frac{\nabla N_t[\mathcal{A}\sharp p] \nabla \mathcal{A}\sharp N_t[p]}{\mathcal{A}\sharp N_t[p]}
\end{aligned}$$

$$
\begin{aligned}
I_2 &= \int N_t[\mathcal{A}\sharp p] \cdot \frac{\partial}{\partial t} \log \frac{N_t[\mathcal{A}\sharp p]}{\mathcal{A}\sharp N_t[p])} \\
&= \int N_t[\mathcal{A}\sharp p] \cdot \left(\frac{\frac{\partial}{\partial t} N_t[\mathcal{A}\sharp p]}{N_t[\mathcal{A}\sharp p]} - \frac{\frac{\partial}{\partial t} \mathcal{A}\sharp N_t[p]}{\mathcal{A}\sharp N_t[p]} \right) \\
&= t \int N_t[\mathcal{A}\sharp p] \cdot \left(\frac{\nabla^2 N_t[\mathcal{A}\sharp p]}{N_t[\mathcal{A}\sharp p]} - \frac{\nabla^2 \mathcal{A}\sharp N_t[p]}{\mathcal{A}\sharp N_t[p]} \right) \\
&= t \int \nabla^2 N_t[\mathcal{A}\sharp p] - \frac{N_t[\mathcal{A}\sharp p]}{\mathcal{A}\sharp N_t[p]} \nabla^2 \mathcal{A}\sharp N_t[p] \\
&= -t \int \frac{N_t[\mathcal{A}\sharp p]}{\mathcal{A}\sharp N_t[p]} \nabla^2 \mathcal{A}\sharp N_t[p], \quad \text{since } \int \nabla^2 N_t[\mathcal{A}\sharp p] = \nabla N_t[\mathcal{A}\sharp p]|_{-\infty}^{\infty} = 0 \\
&= t \int \nabla \left(\frac{N_t[\mathcal{A}\sharp p]}{\mathcal{A}\sharp N_t[p]} \right) \nabla \mathcal{A}\sharp N_t[p] \\
&= t \int \frac{\nabla N_t[\mathcal{A}\sharp p] \nabla \mathcal{A}\sharp N_t[p]}{\mathcal{A}\sharp N_t[p]} - \frac{N_t[\mathcal{A}\sharp p] \nabla \mathcal{A}\sharp N_t[p]^2}{\mathcal{A}\sharp N_t[p]^2}
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial}{\partial t} D_{\mathrm{KL}}(N_t[\mathcal{A}\sharp p] || \mathcal{A}\sharp N_t[p]) &= I_1 + I_2 \\
&= -t \int \frac{\nabla N_t[\mathcal{A}\sharp p]^2}{N_t[\mathcal{A}\sharp p]} - 2 \frac{\nabla N_t[\mathcal{A}\sharp p] \nabla \mathcal{A}\sharp N_t[p]}{\mathcal{A}\sharp N_t[p]} + \frac{N_t[\mathcal{A}\sharp p] \nabla \mathcal{A}\sharp N_t[p]^2}{\mathcal{A}\sharp N_t[p]^2} \\
&= -t \int N_t[\mathcal{A}\sharp p] \left(\frac{\nabla N_t[\mathcal{A}\sharp p]}{N_t[\mathcal{A}\sharp p]} - \frac{\nabla \mathcal{A}\sharp N_t[p]}{\mathcal{A}\sharp N_t[p]} \right)^2 dx \\
&= -t \mathbb{E}_{N_t[\mathcal{A}\sharp p]} \left[\left(\nabla \log \frac{N_t[\mathcal{A}\sharp p]}{\mathcal{A}\sharp N_t[p]} \right)^2 \right] < 0
\end{aligned}
$$

□

D.1 F-LCS Heuristic

Finally, we prove Lemma 2, which provides a necessary-but-not-sufficient condition for F-LCS.

Proof. (Lemma 2) First, note that for scores of any distribution p, and any fixed choice of x,

$$
\begin{aligned}
s^t(x|c_i)[k] &= \begin{cases} \nabla \log p^t(x_{M_i}|c_i)[k], & \forall k \in M_i \\ \nabla \log p^t(x_{M_\ell}|\emptyset)[k], & \forall k \in M_\ell, \quad \ell \neq i \text{ (including } \ell = b) \end{cases} \\
d_i^t(x)[k] &:= s^t(x|c_i) - s^t(x|\emptyset) \\
&= \begin{cases} \nabla \log \frac{p^t(x_{M_i}|c_i)[k]}{p^t(x_{M_i}|\emptyset)[k]}, & \forall k \in M_i \\ 0, \quad \forall k \notin M_i \end{cases} \\
\implies d_i^t(x)^T d_j^t(x) &= 0, \quad \forall i \neq j, \quad \text{since } M_i \cap M_j = \emptyset,
\end{aligned}
$$

where in the second-to-last line we used the fact that the gradient of a function depending only on a subset of variables has zero entries in the coordinates outside that subset. The same orthogonality result also holds for x_0-parametrized networks since the score is related to the conditional mean by $\nabla \log p^t(x_t) := \frac{1}{\sigma_t^2}\mathbb{E}[x_0 - x_t|x_t]$, therefore $v_i^t(x) \propto \mathbb{E}_{p(x_0|x_t,c_i)}[x_0|x_t] - \mathbb{E}_{p(x_0|x_t,\emptyset)}[x_0|x_t]$.

Similarly, we can take an expectation over an arbitrary distribution $x \sim q$ and obtain the following orthogonality result:

$$
\begin{aligned}
\mathbb{E}_{x\sim q}[s^t(x|c_i)][k] &= \begin{cases} \mathbb{E}_{x\sim q}[\nabla \log p^t(x_{M_i}|c_i)][k], & \forall k \in M_i \\ \mathbb{E}_{x\sim q}[\nabla \log p^t(x_{M_\ell}|\emptyset)][k], & \forall k \in M_\ell, \quad \ell \neq i \text{ (including } \ell = b) \end{cases} \\
d_i^t(q)[k] &:= \mathbb{E}_{x\sim q}[s^t(x|c_i)][k] - \mathbb{E}_{x\sim q}[s^t(x|\emptyset)][k] = 0, \quad \forall k \notin M_i \\
\implies d_i^t(q)^T d_j^t(q) &= 0, \quad \forall i \neq j, \quad \text{since } M_i \cap M_j = \emptyset.
\end{aligned}
$$

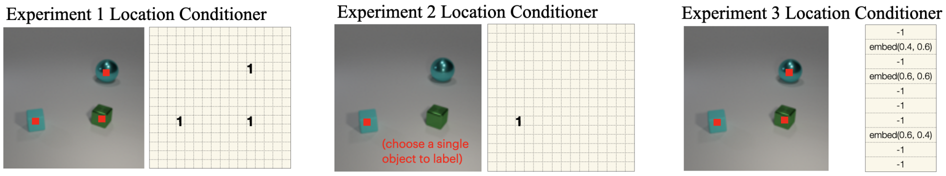
Experiment 1 Location Conditioner
1
1
1
Experiment 2 Location Conditioner
(choose a single object to label)
1
Experiment 3 Location Conditioner
-1
embed(0.4, 0.6)
-1
embed(0.6, 0.6)
-1
-1
-1
embed(0.6, 0.4)
-1
-1

Figure 7: Examples of location conditioning used in Experiments 1, 2, 3 in Figure 1.

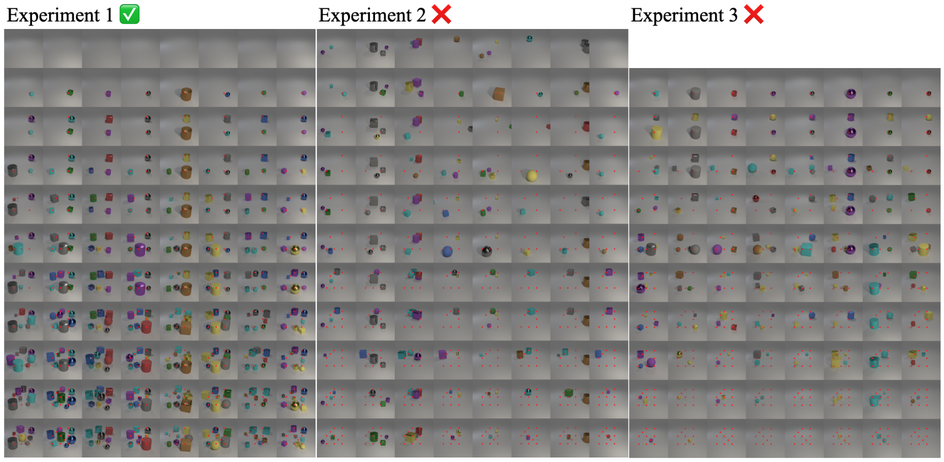

Figure 8: **Length-generalization in Experiment 1, 2, 3 on** $1 - M$ **objects**. We tested length-generalization from $K = 0$ to 10 conditioned locations in each model (each row shows 8 samples for a particular K). (Note Exp. 3 does not support $K = 0$.)

Applying the result to F-LCS scores $s^t_{\mathcal{A}}(z|c) := \nabla_z \log(\mathcal{A}\sharp p)^t(z|c)$, with $q \sim \mathcal{A}\sharp\mathcal{N}(0, \sigma_{t_{\max}})$, at time $t = t_{\max}$, which corresponds to evaluating the score in feature-space at the first denoising step (when the input is Gaussian noise) – gives

$$\begin{aligned} d_i &:= \mathbb{E}_{\eta_{\mathcal{A}}}[s^{t_{\max}}_{\mathcal{A}}(\eta_{\mathcal{A}}|c_i)] - \mathbb{E}_{\eta_{\mathcal{A}}}[s^{t_{\max}}_{\mathcal{A}}(\eta_{\mathcal{A}}|\emptyset)], \quad \eta_{\mathcal{A}} \sim \mathcal{A}\sharp\mathcal{N}(0, \sigma_{t_{\max}}) \\ \implies & d_i^T d_j = 0, \quad \forall i \neq j. \end{aligned}$$

□

E CLEVR: Details and additional experiments

E.1 CLEVR Dataset, architecture, and training details

We used the CLEVR Johnson et al. (2017) dataset generation procedure[3] to generate custom datasets with the default objects, shapes, sizes, colors, but different counts. We generated various datasets with 1 to K objects, with 500,000 samples for each object count, for $K = 1, \ldots, 6$ – for example, models trained on 1-3 objects saw a total of 1,500,000 samples. The image resolution is 128 × 128. Note that objects can interact with each other in this dataset: potentially occluding or casting shadows on each other.

Our experiments cover a few different conditioning setups. Grid-style location-conditioning conditions on 2D object locations, implemented as an 2D integer array representing a `grid_size` × `grid_size` grid over the image recording the count of objects whose center falls within the grid cell, as shown in Figure 7. The count is usually 0 or 1 but can be greater than 1 if object centers happen to land within the same grid cell. We take `grid_size=16` in all experiments. We either record the locations of all objects, or just one of the objects (randomly selected), in the conditioning grid, depending on the experiment.

[3] https://github.com/facebookresearch/clevr-dataset-gen

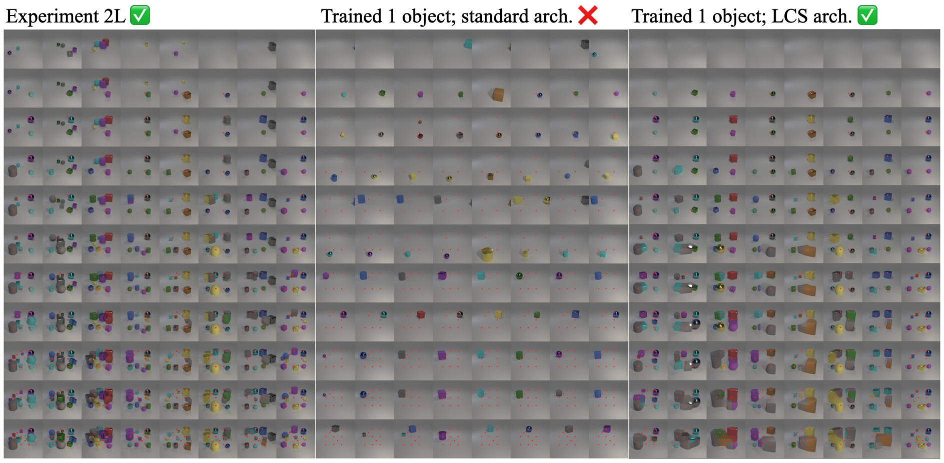
Experiment 2L ✅
Trained 1 object; standard arch. ❌
Trained 1 object; LCS arch. ✅

Figure 9: **Local causal intervention enables length generalization.** (Left) Additional samples from Exp. 2L of Figure 3 (see also Appendix E.3). (Center and Right) Length-generalization in two different location-conditioned models, both trained on images with only a *single* object (and conditioned on its single location). (Center) A model with the standard EDM2 architecture does not length-generalize: it always generates exactly one object (even when conditioned on zero locations). (Right) A model with an explicitly-enforced local architecture as in Exp. 2L length-generalizes up to 6 objects, albeit with some artifacts (objects "merging" into each other). Although it does not perform as well as the Exp. 2L local model trained on 1-3 objects, any length-generalization after training on only one object is remarkable. (In Appendix E.2, we hypothesize that training on more objects, e.g. 1-3, may improve length-generalization by allowing the model to learn *clusters* of objects).

Table 2: **Additional location-conditioning experiments.** In the top table we give the maximum value, $K_{\max}$, such that the model "sometimes succeeds" for every $1 \leq K \leq K_{\max}$, as described in Appendix E.2. Here, $K_{\max}$ is evaluated over 8 samples (vs. 64 in Table 1). In the bottom table, we test $K = 0$ (which is also OOD) and give the range of the number of objects typically produced. Configurations not tested are left blank. Parentheses indicate extra objects at non-conditioned locations, e.g. +(0-2) means there were 0 to 2 extra objects in addition to the K at specified locations. All models are location-conditioned but with different variants: *All labels* means every object was labeled (Experiment 1), *Single label* means only one object (randomly selected) was labeled (Experiment 2), *Rand num labels* means a random number of objects were labeled, and *Drop one label* means all but one object (randomly selected) were labeled.

Train data	All labels	Single label	Rand num labels	Drop one label
1 object	1	1		
1-2 objects	5	1 +(0-1)	3	
1-3 objects	9	1 +(0-2)	5	3 +(0-1)
1-4 objects	10	1 +(0-3)	6	8 +(0-1)
1-5 objects	10	1 +(0-4)	8	9 +(0-1)
1-6 objects	11	1 +(0-5)	9	10 +(0-1)

Train data	All labels	Single label	Rand num labels	Drop one label
1 objects	0-1			
1-2 objects	0	0-2	0-1	
1-3 objects	0	0-3	0-3	0-1
1-4 objects	0	1-4	0-4	0-1
1-5 objects	0	0-5	0-4	0-1
1-6 objects	0	1-6	0-4	0-1

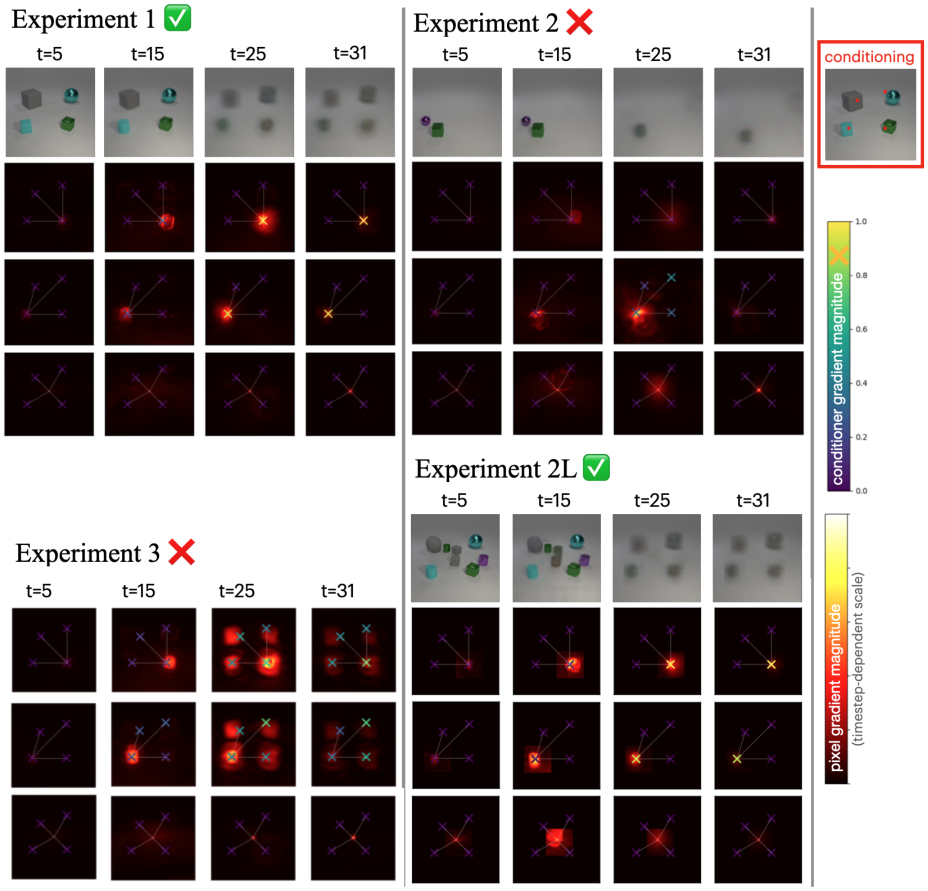

Figure 10: Additional detail for Figure 2. Locality structures in location-conditioned CLEVR models (Exps. 1, 2, 3 of Figure 1 and Exp. 2L of Figure 3). All models are conditioned on 4 locations (OOD). Each column represents a timestep t. Top row shows the predicted denoised images via learned scores. Lower rows (evaluated at two conditioned and one unconditioned location) show heatmaps of the pixel gradient magnitude (average absolute of Jacobian from one pixel to all other pixels), and the conditional gradient magnitude marked with × (with the "gradient" estimated via a finite difference of the score computed with and without each conditioner).

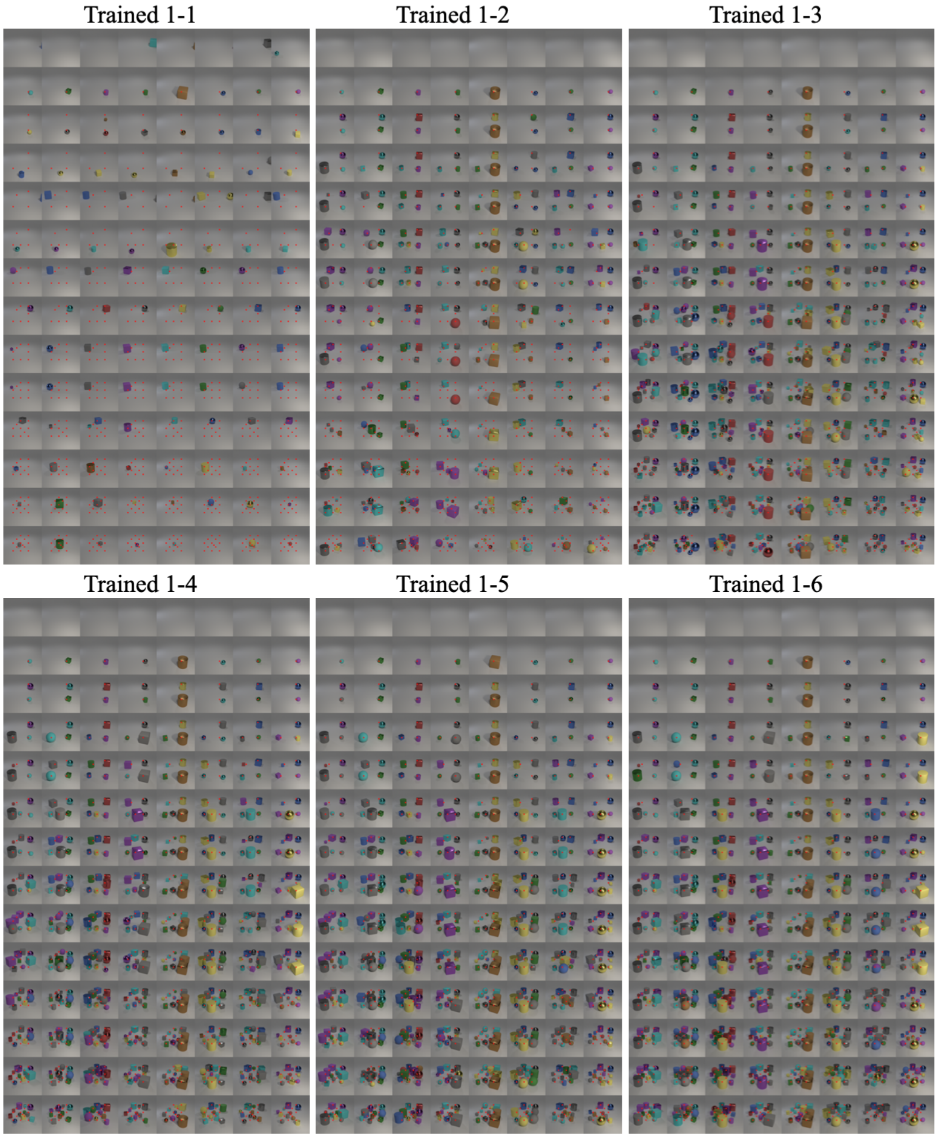

Figure 11: **Length-generalization in Experiment 1 model trained on** $1 - M$ **objects**. We tested length-generalization from $K = 0$ to $K = 12$ conditioned locations in each model (each row shows 8 samples for a particular K).

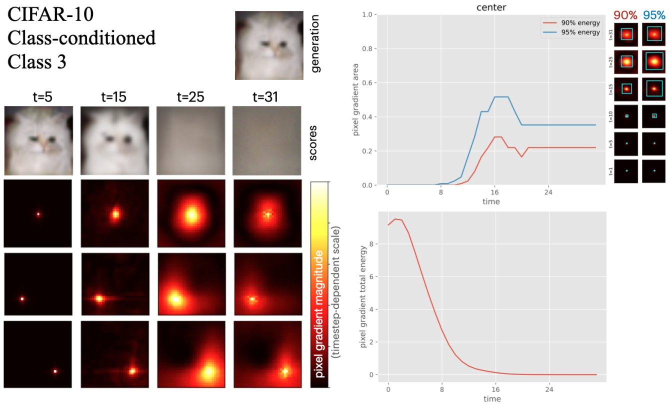

Figure 12: **Pixel locality structure across time for CIFAR-10**, using the same EDM2 architecture from the CLEVR experiments. (Right) Each column represents a timestep t, and the top row shows the predicted denoised images via learned scores. Lower rows show heatmaps of the pixel gradient magnitude (average absolute Jacobian from an selected pixel to all other pixels). Evaluation at three pixel locations confirms prior empirical observations of Kamb & Ganguli (2024); Niedoba et al. (2024) that the effective local neighborhood size is large at high noise levels. (Left) Locality metrics as in Figure 13; details in Appendix F.

List-style location conditioning also conditions on 2D object locations, but lists the (embedded) xy-locations of each object in an array padded with enough slots for up to 10 objects (with each location placed in a randomly chosen slot).

Color conditioning is implemented as a 8-dimensional integer array (there are 8 possible colors) indicating the count of objects with the corresponding color. In all experiments we condition only a single attribute (either location or color) at a time, with all other attributes sampled randomly and not conditioned on.

The 12 locations used for the location-conditioned CLEVR experiments are

```
locations = ([[0.65, 0.65], [0.65, 0.25], [0.25, 0.65], [0.35, 0.35],
[0.45, 0.65], [0.45, 0.25], [0.65, 0.45], [0.25, 0.45],
[0.45, 0.45], [0.55, 0.55], [0.35, 0.55], [0.55, 0.35]]),
```

and the colors are

```
colors = ([blue, brown, cyan, gray, green, purple, red, yellow]).
```

We used our own functionally equivalent re-implementation of the EDM2 Karras et al. (2024) U-net architecture. We used the smallest model architecture, e.g. `edm2-img64-xs` from `https://github.com/NVlabs/edm2`. This model has a base channel width of 128, resulting in a total of 124M trainable weights.

In all experiments, the model is trained with a batch size of 2048 over 128×2^{20} samples, repeating samples if needed. Our training procedure is identical to EDM2 Karras et al. (2024) except that we do weight renormalization after the weights are updated. At inference, we use raw conditional diffusion scores, without applying any guidance/CFG (Ho & Salimans, 2022).

E.2 Length generalization evaluation

In Table 1, we define $K_{\max}$ as the maximum value such that the model "sometimes succeeds" for every $1 \leq K \leq K_{\max}$, with "success" defined as generating K objects at least 25% of the time, and at least $K - 2$ objects at least 90% of the time, with objects appearing in approximately correct locations and with acceptable image quality. This metric is intentionally generous as it is intended to capture the largest K for which the model "sometimes succeeds", rather than requiring perfect performance. To assess these criteria in Table 1, we manually count over 64 samples of each

Experiment	C	0	1	2	3	4	5	6	7	8	9	10
Exp.1, trained 1	1	0	64	0	0	0	0	0	0	0	0	0
Exp.1, trained 1-3	9	0	0	0	0	0	0	1	15	31	17	0
Exp.1, trained 1-5	10	0	0	0	0	0	0	0	0	0	20	44
Exp.2, trained 1	1	0	64	0	0	0	0	0	0	0	0	0
Exp.2, trained 1-3	1	0	64	0	0	0	0	0	0	0	0	0
Exp.2, trained 1-5	1	0	64	0	0	0	0	0	0	0	0	0
Exp.3, trained 1	1	0	64	0	0	0	0	0	0	0	0	0
Exp.3, trained 1-3	3	0	5	13	44	0	0	0	0	0	0	0
Exp.3, trained 1-5	5	0	0	0	5	32	27	0	0	0	0	0
Exp.2L, trained 1	6	0	0	0	0	4	37	23	0	0	0	0
Exp.2L, trained 1-3	9	0	0	0	0	0	0	1	8	21	34	0
Exp.2L, trained 1-5	10	0	0	0	0	0	0	0	0	4	31	29
Color, trained 1	1	0	64	0	0	0	0	0	0	0	0	0
Color, trained 1-3	4	0	0	2	29	33	0	0	0	0	0	0
Color, trained 1-5	7	0	0	0	0	0	6	42	16	0	0	0

Table 3: $K_{\max}$ counts for Table 1 experiments. 'C' indicates the number of locations conditioned on; columns represent number of objects generated at conditioned locations, and rows contain counts of images that contain the number of objects listed in the column.

composition. We test on up to $K = 12$ locations and up to $K = 8$ colors. In Table 3 we provide the complete counts for all experiments shown in Table 1. Table 2 provides a similar analysis for additional experiments with other conditioning configurations such as labeling a random number of objects.

In Table 1 and Figure 11 we observe improvements in length-generalization as we increase the maximum number M of objects the model was trained on, for models that length-generalize at all (i.e. Experiments 1 and 2L). We note that for larger numbers of locations K at inference-time, the objects become crowded (less independent). We hypothesize that in general, models trained on $1 - M$ objects could learn to represent clusters of $1 - M$ objects, as well as how to compose multiple clusters. (This is still consistent with our theory: it is a conditional projective composition where the conditioners are *subsets*. If the underlying data has this type of compositional structure, a trained model could learn to group individual conditioners into subsets in order to exploit it.) If this is the case, a model trained on 1-3 objects could generate, for instance, 12 objects, by composing 4 clusters of 3 objects each. This would mean that models trained on more objects (larger M) could more easily length-generalize to higher K where objects become crowded.

E.2.1 FIGURE 2(RIGHT) DETAILS

In Figure 2 (Right) are a subset of models evaluated in Table 2, each model is shown in a different color (with different shapes indicating different epochs during training). The early, mid, and late epochs are epochs 16777216, 33554432, 134217728, respectively. All experiments used the grid-style conditioner. "All labeled" means every object was labeled (as in Exp. 2), "single object labeled" means only one (randomly selected) object was labeled, and "random number labeled" means that a random number of objects were labeled. After each experiment we include the length-generalization amount at the early, mid, and late epochs.

- brown: trained on 1 object, all labeled (LG 0, 0, 0)
- orange: trained on 1-2 objects, all labeled (LG 0, 1, 3)
- red: trained on 1-3 object, all labeled (LG 3, 3, 6)
- cyan: trained 1-3 objects, single object labeled (LG 0, 0, 0)
- purple: trained on 1-2 objects, random number labeled (LG 0, 1, 1)
- green: trained on 1-3 objects, random number labeled (LG 0, 2, 2)
- blue: trained on 1-5 objects, all labeled (LG 5, 5, 5)

Experiment	C	0	1	2	3	4	5	6	7	8	9	10
trained 1 all label (brown)	1	0	64	0	0	0	0	0	0	0	0	0
trained 1-2 all label (orange)	5	0	0	0	5	21	38	0	0	0	0	0
trained 1-3 all label (red)	9	0	0	0	0	0	0	1	15	31	17	0
trained 1-3 single label (cyan)	1	0	64	0	0	0	0	0	0	0	0	0
trained 1-2 rand label (purple)	3	0	0	2	62	0	0	0	0	0	0	0
trained 1-3 rand label (green)	5	0	0	0	0	3	61	0	0	0	0	0
trained 1-5 all label (blue)	10	0	0	0	0	0	0	0	0	0	20	44

Table 4: $K_{\max}$ counts for Figure 2 (Right) experiments. 'C' indicates the number of locations conditioned on; columns represent number of objects generated at conditioned locations, and rows contain counts of images that contain that number of objects listed in the column.

In Figure 2 (Right), the x-axis shows length-generalization, defined as the number of locations to which the model can generalize *beyond* the number on which it was trained (e.g. +6 for a model trained on 1-3 locations that generalizes to 9). The number of locations to which the model can generalize is evaluated via $K_{\max}$ as described in Appendix E.2, with the complete counts given in Table 4. The conditional locality (y-axis) metric is described in Appendix F.

E.3 Details of Experiment 2L: Local Patch-Based Architecture Intervention.

For Experiment 2L (Figure 3 and 9), we developed a local variant of the EDM2 model that processes images as a grid of overlapping patches. The image is divided into a `grid_size × grid_size` grid of cells, where each cell has size `m = resolution / grid_size`. We set `grid_size = 16` to match the location-conditioning grid. For each grid cell at position (i, j), we extract a patch of size $M = (2k + 1) \times m$, where k is the neighborhood radius. We used $k = 2$ for the experiments in this paper. Each patch is conditioned only on the location-conditioners that fall within the patch – that is, a $(2k + 1) \times (2k + 1)$ subgrid of the full conditioning grid. Each patch is also conditioned on its absolute location (that is, the model is not equivariant). This is implemented by appending the patch center coordinates i, j to the flattened location-conditioner grid to form the complete conditioner. The training procedure uses a patch sampling approach balancing positive and negative examples for efficiency. For each training image, we randomly sample two patches: one "positive" patch with at least one active conditioner, and one "negative" patch with no active conditioners. Standard EDM2 loss is applied to each patch and losses are averaged.

At inference, we reconstruct the full image by processing each grid cell as follows: extract the corresponding noisy patch and local conditioner; denoise the patch using the trained local model; copy the center region (the single cell) of the denoised patch back to the full image.

In Experiment 2L, we trained the local model on the same dataset and location-conditioning (labeling only a single object) as in Exp. 2, showing that Exp. 2L length-generalizes while Exp. 2 fails. As a test, we also verified that setting $k = 8$ (which for grid size 16 makes the patches the size of the image) reproduces the behavior of Exp. 2 (i.e. length-generalization fails). We also trained the local model on a dataset with only a single object per image and show that it length-generalizes up to 6 objects in this case, whereas a standard model trained on only one object per image only generates one object per image at test time, regardless of conditioning (Figure 9).

E.4 Experiment 3L - Preliminary

Analogous to Experiment 2L, we study a local version of the non-length-generalizing Experiment 3 and show preliminary evidence of length-generalization for the local model, as shown in Figure 14. In Experiment 3L, the model denoises individual patches conditioned only on the location-conditioners that fall within the patch, with conditioners represented as a list as in Experiment 3. The setup is as described in Appendix E.3, with the only difference being the list-style conditioner. Specifically, the original conditioner lists the locations of each object in an array padded with enough slots for up to 10 objects (with each object placed in a randomly-chosen slot); the patch conditioner includes only the conditioned locations that fall within the current patch, re-centered relative to the patch center, and placed within the padded array in their original random slots. Each patch is also conditioned on its absolute location as described in Appendix E.3. The current results are shown at

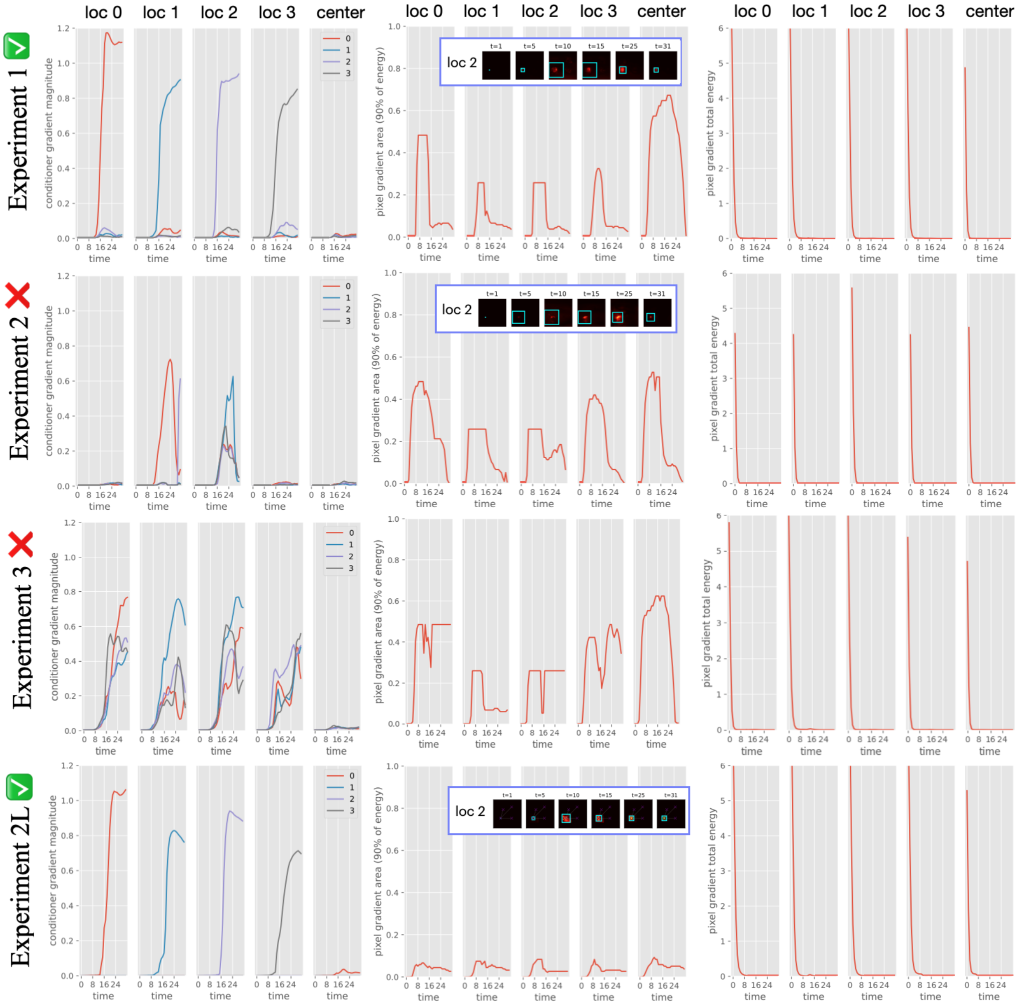

Figure 13: **Locality metrics for CLEVR length generalization** for Experiments 1, 2, 3, and 2L, conditioned OOD on 4 locations, as in Figure 2. (Left) Conditioner gradient magnitudes at selected pixels (loc 0–3 and center), with colors indicating individual conditioners. Experiment 1 shows strong conditional-locality at high noise (each pixel responds only to its corresponding conditioner), whereas Experiment 2 and 3 exhibit non-local responses. Experiment 2L has conditional-locality explicitly enforced by the architecture. (Middle) Pixel gradient area required to cover 90% of score gradient "energy" (sum of squared magnitudes). Insets illustrate the selected square regions at loc 2 at a few timesteps. We observe similar pixel-locality between Experiments 1 and 2; the pixel gradients are highly localized at both high and low noise, but delocalize during intermediate timesteps. Experiment 3 exhibits pixel non-locality even at high noise. Experiment 2L has pixel-locality explicitly enforced by the architecture. (Right) Total pixel gradient energy (over entire image), which is higher at low noise levels, consistent with conditioners dominating the score field at high noise and pixel interactions emerging later in denoising.

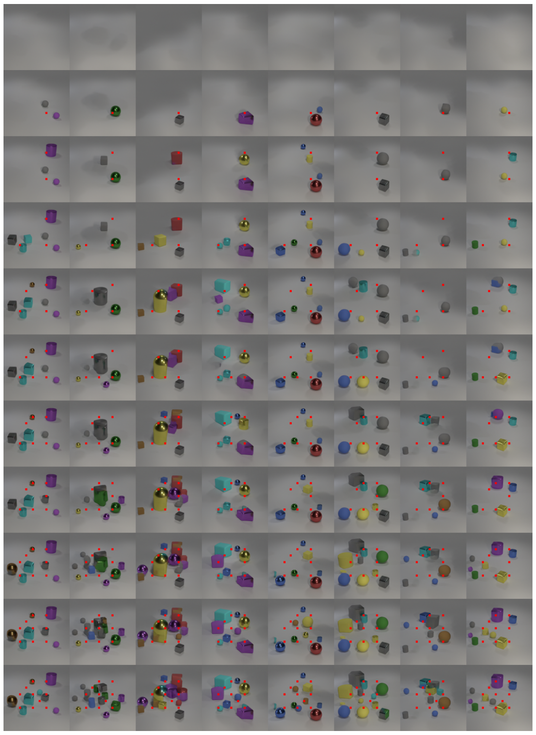

Figure 14: Experiment 3L - Preliminary.

an early checkpoint during training – we did not have enough time for the training to finish and we expect the location-accuracy to improve with more training – but the initial results are promising, showing significant length-generalization (Figure 14). We will complete the experiment and provide a more thorough discussion and metrics for camera-ready should the paper be accepted.

E.5 Detail of color-conditioned CLEVR feature-space study.

In Figure 4, the similarity matrices show cosine similarities (heuristic Lemma 2) between network layer activations, when the model is conditioned on different colors (that is, the i, j entry is the cosine similarity between the mean difference vectors for colors i and j). We perform the study on MCScale layers, which play a similar role to cross-attention in EDM2. Figure 4 shows the encoder MCScale layers only; Figure 15 provides the complete length-generalization plots and cosine similarities between colors for all EDM2 MCScale layers. We identify a possible F-LCS feature-space within the early encoder layers: specifically the layers with resolution 128 and 64 (in the figure, the average of these layers is labeled the “feature-space”). Note that this is an empirical and subjective identification, based on our observation that the cosine-similarity heuristic suggests F-LCS within these particular layers; we currently have no theoretical basis for predicting whether and where such structure might occur within the network. Nevertheless, the fact that F-LCS structure does seem to appear *anywhere* within the network’s internal representation helps to explain the observed partial length generalization according to our theory.

F Pixel- and Conditional-Locality Gradient Analyses

In this section we describe the pixel- and conditional-locality analyses used in Figures 2, 5, 12, 13.

Pixel-locality is measured as pixel gradient magnitude (average absolute of Jacobian from one pixel to all other pixels), and conditional locality is measured as conditional gradient magnitude (with the “gradient” estimated via a finite difference of the score computed with and without each conditioner). Further detail follows.

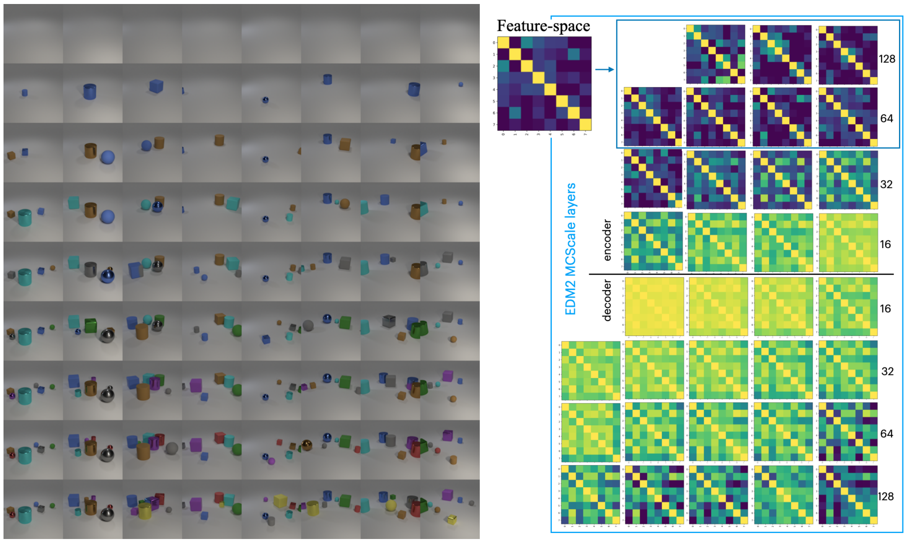

Figure 15: Additional detail for Figure 4 color-conditioned CLEVR study. Complete length-generalization plots (1 to 8 colors; > 5 colors is OOD) and cosine similarities between colors for all EDM2 MCScale layers (both encoder and decoder).

Pixel Gradient Magnitude We follow Niedoba et al. (2024) in measuring the average absolute of Jacobian from one pixel to all other pixels. The gradient computation uses automatic differentiation to compute the derivative of each output channel at the target pixel with respect to all input pixels. Specifically, for a model output $\widehat{x}_0 = f_\theta(x_t, t, c)$ with shape $[B, C, H, W]$, we compute:

$$G_{i,j}(x, y, t) = \frac{1}{B} \sum_{b=1}^{B} \sum_{c=1}^{3} \left| \frac{\partial \widehat{x}_0[b, c, y, x]}{\partial x_t[b, c, i, j]} \right| \tag{8}$$

where (x, y) is the target pixel and (i, j) ranges over all input pixels. The spatial gradient map $G \in \mathbb{R}^{H \times W}$ indicates how strongly each input location influences the prediction at the target pixel.

Conditional Gradient Magnitude To measure the influence of each conditioning label on the prediction at a target pixel location, we approximate the "conditional gradient" via a finite difference by computing output differences when individual labels are ablated from the conditioning set.

Given a conditioning set $C = \{c_1, c_2, \ldots, c_K\}$, we compute the influence of condition c_k at target pixel (x, y) as:

$$I_k(x, y, t) = \|f_\theta(x_t, t, C)[x, y] - f_\theta(x_t, t, C_{-k})[x, y]\|_2 \tag{9}$$

where C_{-k} represents the conditioning set with label k removed, and the norm is taken across color channels. The method is implemented using forward passes only (no actual gradients are required).

Locality Metrics In Figure 13 we directly plot the gradient influence $I_k(x, y)$ for each conditioner k across all times t. We quantify spatial locality by finding the smallest square centered at each target location that contains a specified fraction (90%) of the total gradient energy, i.e. the smallest square s s.t. $\sum_{(i,j) \in \text{square}_s} G_{i,j}^2 \geq 0.9 \times \sum_{i,j} G_{i,j}^2$.

In Figure 2 (Right), Conditional Locality is an aggregated locality metric obtained by calculating the conditional gradient magnitude at each conditioned location, and computing

$$\text{Conditional locality} = \frac{\sum_k I_k(x_k, y_k, t)}{\sum_k \sum_{k'} I_k(x'_k, y'_k, t)}.$$

G SDXL EXPERIMENT DETAILS.

For Figure 5 we use a pretrained SDXL model (out-of-the-box, no finetuning), specifically `stabilityai/stable-diffusion-xl-base-1.0`, with the prompt "a beautiful photograph with a horse in the middle, a dog on the left, and a cat on the right" – which contains implicit location conditioning. We perform a locality analysis similar to the one we used for CLEVR as described in Appendix F, but with a few adaptations. The gradient computation is performed in the VAE latent space. Since SDXL uses more complex conditioning, the conditioning influence analysis takes $C = \{\text{text\_embeds}, \text{pooled\_embeds}, \text{time\_ids}\}$ and $\mathcal{C}_{-k}$ represents the conditioning with word k removed from the text prompt. Specifically, we first first tokenize the input prompt into individual words, remove common words like "a", "the", etc., and for each remaining word creating a modified prompt by removing that word. We then compute the score difference w.r.t. the modified prompt. We rank the words by their influence magnitude to study how specific words affect the score at particular pixel locations. We consider a set of words "dominant" if their minimum influence magnitude is at least 2x the influence of the next-ranked word. We ran the model with 50 inference steps using the standard `EulerDiscreteScheduler` stopped at steps $t = 1, 25, 50$.

For Figure 6 we used the SDXL model (`stabilityai/stable-diffusion-xl-base-1.0`) as above. We used the following prompts for the feature-space analysis (listed with the shorthand used in the figure):

- dog: "A dog, full body, highly detailed photograph."
- horse: "A horse, full body, highly detailed photograph."
- cat: "A cat, full body, highly detailed photograph."
- vangogh: "An oil-painting in the style of Van Gogh."
- monet: "A watercolor-painting in the style of Monet."
- hokusai: "A woodblock print in the style of Hokusai."
- sushi: "Eating sushi with chopsticks, highly detailed photograph."
- croissant: "Eating a croissant, highly detailed photograph."
- vangogh+cat+sushi: "A cat eating sushi with chopsticks, oil-painting in the style of Van Gogh."
- vangogh+cat: "A cat, full body, oil-painting in the style of Van Gogh."
- sushi+cat: "A cat eating sushi with chopsticks."
- unconditional: "",

We used the `attention_map_diffusers` library wooyeolbaek (2025) to hook into SDXL's cross-attention layers within the Down, Mid, and Up blocks. We ran with 15 inference steps total with the standard `EulerDiscreteScheduler` but extracted attention maps at intermediate times (step 1, 7, and 15 for low, mid, and high noise). To approximate the score difference vectors for heuristic 2 we generated 10 samples per prompt and averaged the cross-attention maps, subtracting the unconditional average from each prompt average. Importantly, for each prompt, we actually selected only the specific cross-attention map corresponding to the *single token* most relevant to the concept (namely: dog, horse, cat, gogh, monet, sai, sushi, croissant), and for the unconditional prompt we selected the end-of-sequence token (which we found received most of the attention). We then computed cosine similarities between the mean difference vectors for each pair of prompts. Figure 16 shows a larger version of the cosine similarity heuristic for all cross-attention layers shown in Figure 6, as well as the mid block activations at low and mid noise levels. We identify a possible structured feature-space within the first 5 cross-attention layers of the Mid block; note that this an empirical and subjective identification, but may help explain the model's observed compositional success (see discussion in Appendix E.5).

For the Compositional attention analysis, we use the composite prompts listed above and compute the cosine similarity between the mean difference vector for the compositional prompt (e.g. vangogh+cat+sushi) evaluated at a specific token (e.g. cat) and the mean difference vector for the single "cat" prompt, evaluated at the cat token.

H Testing OOD prompts with a model trained on Flikr

Since the train sets of many large-scale text-to-image models like SDXL are not publicly known, the extent to which they are truly capable of OOD generalization is unclear. We therefore study a

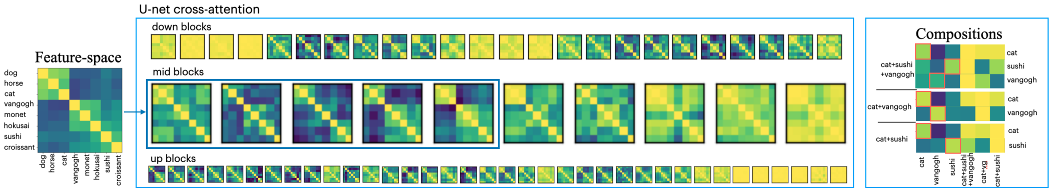

Figure 16: Additional detail for Figure 6. (Left) Cosine similarities between selected concepts for all SDXL cross-attention layers. The mid block activations at low and mid noise levels are also shown; note that the disentanglement is weaker at low noise. Disentanglement is evident in the early mid blocks at high noise (and also appears in some later down blocks and earlier up block), suggesting these layers could serve as a compositional feature-space. (Right) *Compositions* show that within multi-concept prompts (e.g. cat+sushi+van Gogh), individual token (e.g. cat) activations have high similarity with corresponding single-concept (cat) activations, consistent with compositional structure.

model for which the training set is known: Gu et al. (2023a)'s Matroyshka Diffusion Model (MDM) trained on subset of 50M Flikr images. In order to study OOD generalization, we designed candidate prompts and searched through the training set captions for keyword matches. We found some evidence of OOD composition on prompts with no conceptually-similar counterpart in the training set. Notably, we found that "a cat eating sushi" actually *does* actually appear in the train set;[4] however, other similar prompts such as "a dog eating a croissant" or "a horse eating sushi" do not appear. One OOD example was "a cow jumping over a candlestick": no conceptually similar prompts were found in the train set[5], and yet the model is able to produce some plausible samples.

In Figure 17 we show several OOD example generations from MDM, as well as a feature-space analysis; we compare this to SDXL (although it is not known if the prompts are OOD for SDXL). We used the following prompts: "a cow jumping over a candlestick," "a watercolor painting of a cow jumping over a candlestick," "a dog eating a croissant," "a watercolor painting of a dog eating a croissant," "an oil-painting of an octopus flying through outer space,", "an oil-painting of cat eating sushi with chopsticks." (Only the last prompt is in-distribution for the Flikr training set; for all other compositional prompt a keyword search of the train set found no conceptual matches). MDM is evidently capable of some degree of compositional generalization, generating at least some plausible samples for the OOD prompts. The quality and prompt-fidelity of SDXL is higher, though we do not know whether the prompts are OOD for SDXL.

Figure 17 also shows feature-space cosine similarity experiments, as described in Appendix G. We added the following single-topic prompts: "A(n) cow/octopus/candlestick, highly detailed photograph"; also, since Van Gogh and Monet do not appear in the MDM Flikr train set, for this study we instead use "An oil-painting/watercolor of a landscape". For SDXL we select the cross-attention maps for the tokens (dog, cat, cow, octopus, oil, watercolor, jump, air, space, stick, croissant), while for MDM we always select the final token since empirically it receives by far the most weight. The experiments show some degree of concept disentanglement within the mid block learned feature spaces, though less clearly in MDM than in SDXL (however, MDM only has a single mid block, and there also appears to be some disentanglement within its other layers). These experiments offer preliminary – though far from conclusive – evidence that OOD compositional generalization is actually possible, and that compositional structure in feature-space may support it.

[4] A train-set caption conceptually matching cat+sushi is: "A room with a wall painted with a mural of a cat eating sushi. The wall has a banner at the top with the words "DIADEMANG" written on it. The room has two low tables with cushions on the floor, and plates of sushi are placed on the tables. The lighting in the room is dim."

[5] Two train-set captions matching the keywords "cow" and "candlestick" were found, for example: "A page from an old book with various crests, including one with scissors and a scissor-like symbol, one with a cow, one with a candlestick, one with a statue of a man holding a heart, and one with a shield with a cross and the words 'La Comte Des Marechaux.'..." but neither conceptually represented a cow jumping over a candlestick.

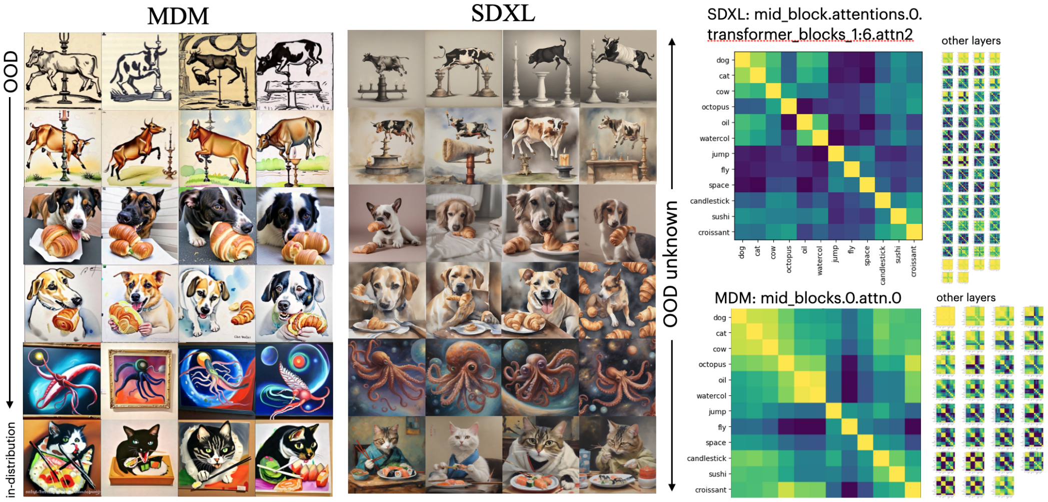

Figure 17: **MDM generalization on OOD prompts**