FlowLess: Controlling Abstract Image Generation

Amir Hertz✉ and Noah Snavely

Google DeepMind

Abstract. While large-scale text-to-image models excel at generating high-fidelity content from semantic descriptions, users often lack precise control over the of non-semantic *abstract* elements and the model's expressive freedom. Existing structural conditioning methods typically impose rigid spatial layouts or global style constraints, failing to address the need for fine grained creative control via mixture of visual primitives. We introduce a novel self-supervised framework that enables granular control over image generation through a *visual abstraction set*—a collection of disjoint shapes, textures and composition details. By fine-tuning a pre-trained flow model, we allow users to construct diverse semantic concepts from the same underlying visual kit. Our method offers control over the *level of abstraction*: by leveraging a training strategy based on varying geometric augmentations and per-part modulation, users can dictate whether the model should preserve input shapes or interpret them loosely to fit the narrative. Furthermore, we enable control over the generation's expressiveness, allowing for the creation of minimal compositions. We demonstrate that our method provides a richer, more flexible paradigm for creative design compared to state-of-the-art baselines across diverse styles and compositions.

Keywords: abstraction control · image generation

1 Introduction

> *Beauty is that reasoned harmony of all the parts within a body, so that nothing may be added, taken away, or altered, but for the worse.*
>
> Leon Battista Alberti, 1485

In his classic ten-book treatise on architecture, Alberti considers beauty as the building's most noble attribute; an inherent property of the body itself, distinct from *ornament*, which he characterizes as an auxiliary attachment or complement to the form [2]. Alberti's sentiment has resonated throughout art history, extending from the Renaissance to the present day. In particular, it prefigured the ethos of modern movements [5,28,43], characterized by a drive toward ornament-free abstraction where essential elements are locked in a balance from which nothing can be removed [16].

Modern large-scale text-to-image (T2I) generative models [32, 39, 45] have demonstrated remarkable capabilities in synthesizing high-fidelity images that

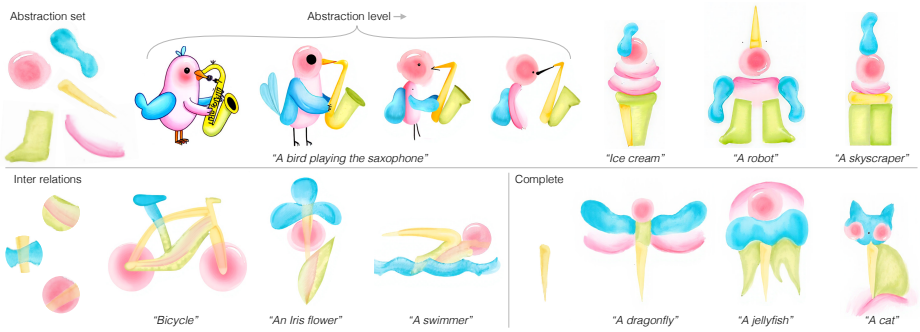
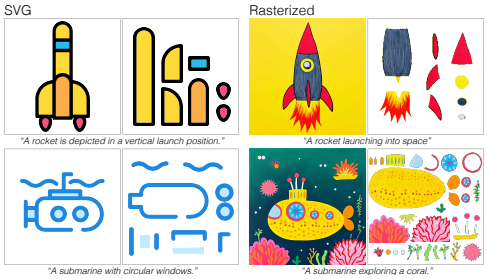


Fig. 1: Controlling abstract image generation. Our method enables image generation of different concepts from a provided visual abstraction set. Moreover, we provide further fine-grained control over the abstractness level of the generated image (top row), and over the composition of the visual elements using additional provided visual details (bottom row).

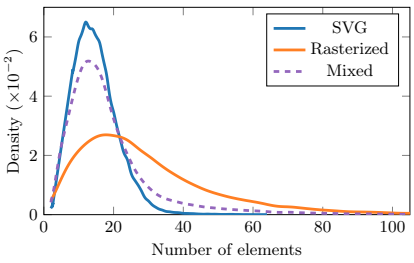
adhere to input text descriptions. However, achieving the "reasoned harmony" Alberti describes remains a significant challenge. Beyond the difficulty of controlling individual visual parts and their interrelations, users distinctively lack the ability to govern the model's *expressiveness*. Current methods do not allow the user to specify whether the result should embody a minimal form, free from the "ornaments" inherent to generative priors—or, conversely, if the model should be granted the freedom to elaborate and fill the canvas. While text prompts provide high-level semantic guidance, they fail to specify this desired granular level of abstraction. Similarly, existing visual conditioning methods typically operate on distinct semantic entities [22, 60, 62] or impose global style and spatial conditions [17, 54, 64, 65] yet they lack the ability to flexibly compose and harmonize *abstract* visual inputs or explore novel visual expressions.

We introduce *FlowLess*, a novel framework that moves beyond traditional style transfer by utilizing a *visual abstraction set*. By guiding a *flow* model [27] to rely solely on this set, we ensure it generates *less* unnecessary visual clutter by strong generative priors. Our method enables the synthesis of diverse semantic concepts from the same underlying visual elements; for instance, the different concepts generated from the watercolors shapes in Fig. 1 (top). Furthermore, FlowLess allows users to control the *level of abstraction* with respect to these elements, as demonstrated by the varying abstractness of the "bird–saxophone player" example in that figure. Users can also guide the generation process by providing visual details that illustrate *relations* between elements, for example, the transparent overlay relation shown in Fig. 1 (bottom). Finally, FlowLess supports image completion, allowing users to provide an initial drawing to guide the generation of specific content, such as the various animals drawn around a fixed yellow triangle in Fig. 1 (bottom).

Our method employs a self-supervised training scheme to fine-tune a flow model [23], utilizing a simple pipeline that decomposes training images into a



(a) Training examples from the SVG dataset (left) and the rasterized dataset (right).



(b) Distribution over the number of elements in the training examples' abstraction sets.

Fig. 2: FlowLess dataset mixture. Our training examples includes pairs of images and corresponding decomposed elements derived from both SVG vector images and raster images **(a)**. The number of elements varies across different samples; SVG images tend to consist of lower number of elements **(b)**.

visual abstraction set comprised of disjoint elements and a complementarity set of composition details that specify the relation between elements. To achieve granular control, we introduce a training strategy based on varying *levels of augmentation* and random element dropout. We subject the decomposed elements to a variety of augmentations while simultaneously varying the proportion of the original set provided to the model. By mapping the augmentation level and coverage of the input elements into the modulation layers of the flow model, the model implicitly learns to interpret visual inputs with varying degrees of freedom and expressiveness.

In summary, we propose a self-supervised framework for abstract image generation that enables granular control over image composition without the need for manual annotated supervision. By integrating abstraction sets directly into the flow model via a per-part modulation scheme, FlowLess allows for mixed-constraint generation, enabling users to rigorously preserve specific shapes while allowing the model to interpret others more freely. Our approach enables flexible, detailed creative control, enabling users to move beyond strict layouts and instead guide the image generation through abstract visual elements and details. We demonstrate the effectiveness of our method through extensive qualitative results and quantitative comparisons to state-of-the art stylization and context aware image generation methods.

2 Related Works

Text-to-Image generation and editing. Diffusion and flow-based models [23, 39, 42, 45] have demonstrated remarkable capabilities in generating high-fidelity images from textual descriptions. Consequently, these models have become the core engine for diverse creative tasks, including masked editing [4, 32, 55], spatially conditioned generation [52, 64] and others [14, 15, 29, 56]. While recent large-scale

models [7, 24, 33, 58] offer versatile editing capabilities through multi-task contextual training, they often struggle with abstract representations that deviate significantly from their training distribution.

Compositional image generation. Compositional generation from disjoint visual elements or concepts has been studied in several adjacent forms, particularly focusing on extracting and composing different semantics from various input images. A growing body of work achieves unified semantic composition and control through image conditioning [8, 9, 30] or learned concept tokens [3, 13, 31, 51]. More closely related to our work is *Piece-it-Together* [41], which fine-tunes a diffusion model to compose specific object categories, such as imaginary characters or products. While differing inputs and objectives mean none of these are direct baselines for our approach, they highlight the vast potential of part-based generation. Our framework extends this paradigm by composing disjoint *abstract* elements rather than explicitly defined semantics.

Style control. Our work is also related to exemplar-based stylization. Since paired datasets (original vs. stylized) are generally unavailable, stylization research typically relies on unsupervised or self-supervised frameworks. Unsupervised methods [17, 44] often employ shared attention mechanisms [1] to inject reference style features into the generation process. Alternatively, self-supervised approaches fine-tune low-rank adapters [19] on small datasets [46], or on a few examples of a specific style [54, 63]. Our approach follows a similar self-supervised logic by fine-tuning a flow model where both the condition and the ground truth are derived from the same sample (see Fig. 2a). However, unlike traditional stylization, our framework provides granular control over individual abstract elements, enabling a higher degree of creative freedom and abstraction.

Abstract image generation. Abstraction in visual synthesis involves the strategic simplification of form while maintaining semantic essence. Early work in this domain, such as CLIPDraw [12] and VectorFusion [21], demonstrated that complex concepts could be synthesized by optimizing sparse geometric primitives via differentiable vector graphics [25]. More recently, abstraction control has been explored in specialized creative tasks. Word-As-Image [20] applies semantic abstraction by morphing letter forms into representative shapes; IconShop [59] focuses on the generation of stylized, abstract icons; and DeepVecFont [57] addresses the synthesis of minimalist typographic forms. While these methods achieve impressive results within their respective domains, they often rely on domain-specific training or per-instance optimization. In contrast, our framework introduces a generalizable *visual abstraction set*. Rather than optimizing fixed paths or restricting domains, our method interprets disjoint visual elements with varying degrees of freedom, harmonizing them into a versatile creative design.

3 Preliminaries

Rectified Flow. Flow-based text-to-image generative models [26, 27] are trained to construct a velocity function v_θ between a random known distribution X_1, e.g., Gaussian noise, to the unknown distribution of natural images X_0, using the following loss:

$$\mathcal{L}_\theta = \mathbb{E}_{t,x_0,p,x_1}\left[\|v_\theta(x_t,t,p) - (x_1 - x_0)\|^2\right], \tag{1}$$

where $t \in [0,1]$ is sampled according to a training scheduler function (e.g. Cosine), p is a text prompt corresponds with the sampled image $x_0 \sim X_0$, $x_1 \sim \mathcal{N}(0,I)$ is a sampled noise and x_t is given by $x_t = (1-t)x_0 + tx_1$.

The inference sampling process, starting from $x_1 \sim \mathcal{N}(0,I)$ and conditioned on text p, is given by integrating the predicted velocity:

$$x_0 = x_1 + \int_{t=1}^{0} v_\theta(x_t,t,p)dt \tag{2}$$

using any ordinary differential equation solver (e.g., an Euler solver).

In practice, many flow-based generative models, including the one we use [23], operate within a compressed latent space [42]. During training, x_0 is encoded from a real RGB image, and during inference, the final image is generated by decoding the sampled x_0 using a pre-trained auto-encoder.

Diffusion transformer (DiT). The DiT architecture [35] has become a popular choice for diffusion and flow-based generative models [11, 23, 53]. A DiT jointly processes a sequence of text and image embeddings, denoted by $Z = \{z^i_{\text{text}}\}_{i=1}^{\#\text{text}} \cup \{z^i_{\text{image}}\}_{i=1}^{\#\text{image}}$, where #text is the number of text embeddings, encoded from p [37], and #image is the number of image embeddings, patchified [10] from x_t. In addition to the to standard transformer's attention and feed-forward (FF) layers [50], the DiT block includes modulation layers that incorporate an additional, non-sequential modulation embedding $y = MLP(t,c)$, where c is the CLIP pooled embeddings [36] of the input text p.

Modulation layers. Each modulation layer regress a scale vector $\alpha = MLP_\odot(y)$ and a shift vector $\beta = MLP_+(y)$ which are applied per-channel over an intermediate activations sequence: $\hat{z}_i = \alpha \odot z_i + \beta$. These modulation layers are applied before and after each attention and FF layer.

Modulation space. Adding residual directions to the modulation vector, $y + \Delta_{\text{attribute}}$, has been shown to provide semantic control over generated images [13]; carefully selected directions can guide the generation process toward specific attributes, such as identity, style, or pose. Moreover, learning distinct directions for different members of the input sequence results in a disentangled representation: formally, the per-embeddings modulation is given by $\hat{y}_i = y + \Delta_{\text{attribute}}$ where $\hat{y}_i$ is used to modulate z_i. Applying different scales and shifts across specific sequence embeddings can affect individual elements within the image or

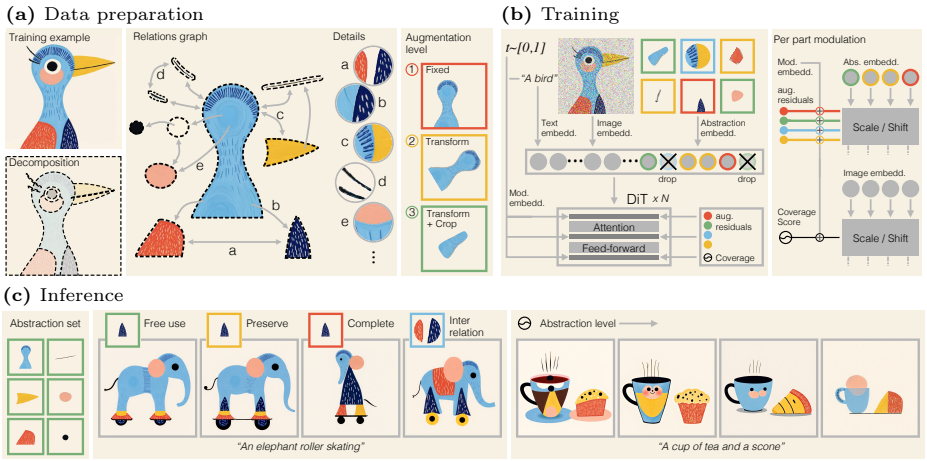


Fig. 3: Method Overview. We extract a visual abstraction set comprising disjoint elements and inter-relation details from real training images, which undergo varying levels of geometric augmentation **(a)**. During training, we fine-tune a flow model by passing these augmented elements into a DiT alongside standard text and image embeddings; each augmentation type dictates a learned residual vector that modulates its corresponding abstraction embedding, while a global coverage score modulates the image embeddings **(b)**. Finally, during inference, the user can define the specific generative role of individual abstraction elements (e.g., free use, preserve, or complete) and adjust the overall abstraction level to constrain the model's expressiveness **(c)**.

provide global, multi-attribute control. Specifically, the effectiveness of learning per-embedding modulation has been demonstrated through the manipulation of text embeddings [13].

4 Method Overview

We propose a self-supervised training framework for a pre-trained text-to-image flow model that conditioned the image generation process on an additional visual abstraction set comprised of non-semantic elements. In the following sections, we detail our data preparation pipeline, the conditioning modifications introduced during training, and the methods for controlling image generation via these abstractions during inference.

4.1 Data preparation

The objective of our data preparation pipeline is to pair training images with a visual abstraction set containing non-semantic elements that collectively define the image's appearance. This set may include textures, shapes, spatial relationships between elements, or any other visual elements that provide information

about the appearance of the image without revealing semantic content. Since such paired data is expensive to obtain, we propose an auto-decomposition and augmentation pipeline to extract a proxy for such abstraction sets.

Abstraction set extraction. Given an input training example, we first decompose it into individual elements. We handle two image formats: i) SVGs which inherently store decomposition information as paths or shape primitives, and ii) standard rasterized images, which require a more involved decomposition process using an off-the-shelf segmentation model [40] followed by an inpainting model [23] to fill occluded regions. The number of decomposed elements per training example is summarized in Fig. 2b, and individual decomposed examples are shown in Fig. 2a. Note that while rasterized images feature more diverse artistic styles and textures compared to vectorized images, they typically consist of more elements, and their decomposition is often suboptimal. For instance, the flames element of the rasterized rocket (Fig. 2a right) reveal semantic information, and the rasterized submarine is possibly over-decomposed, whereas the SVG decompositions are of higher quality. We later ablate the importance of each dataset, see Sec. 5.2.

After obtaining the set of disjoint elements, we construct a relation graph where each edge e_{ij} represents an inclusion, intersection, or proximity relationship between two elements, as illustrated in Fig. 3a. We render each inter-relation edge as a single close-up *detailed* view, serving as a visual proxy for the relationship between the two parts.

Level of augmentation. To enrich our abstraction proxy and mitigate the effects of imperfect decomposition, we augment each part in the set using one of three increasing levels of distraction (Fig. 3a). In the lightest , we provide the segment at its original *fixed* location and scale. In the intermediate level, we apply random rotation and scaling, and place the segment in the center of its layout. In the most aggressive augmentation, we further omit the segment's contour information by applying a random crop. The final abstraction set includes four types of elements: the three augmentation levels and the complementary extracted relation close-ups. More implementation details are provided in Section 5.1.

4.2 Training

We fine-tune the Flux.1–dev DiT model [23] over extended sequences that include embeddings of the abstraction set, given by:

$$Z = \{z^i_{\text{text}}\}_{i=1}^{\#\text{text}} \cup \{z^i_{\text{image}}\}_{i=1}^{\#\text{image}} \cup \{z^i_{\text{abs}}\}_{i=1}^{\#\text{abs}}, \tag{3}$$

where #abs is the number of embeddings encoded from the abstraction set (see colored outline circles in Fig. 3b). The loss function in Eq. 1 remains unchanged.

Per-part modulation. To incorporate the different abstraction types associated with the elements in the set we optimize modulations residuals parameters. These

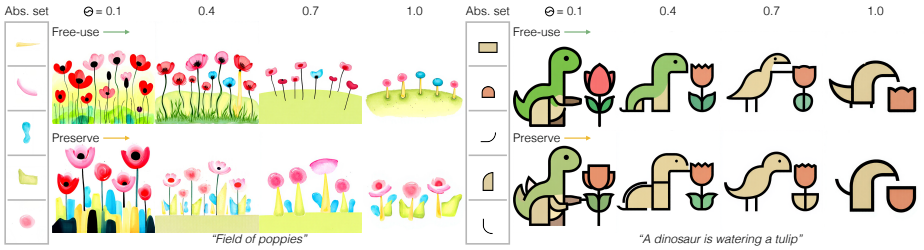


Fig. 4: Level of abstraction. By increasing the coverage, from left to right, we reduce the number of elements in the generated and images, constraining the model to use only the abstraction set input (left columns).

residuals are added in a per-embedding manner (as described in Sec. 3): $\hat{y}_i = y + \Delta_{\text{aug}}$, where Δ_{aug} are the learned residuals associated with the augmentation type applied to element i. This operation is illustrated in Fig. 3b (right), where augmentation modulation residuals affect to the corresponding outline color-matched abstraction set embeddings in the modulation layers.

Coverage score. As shown in Fig. 2, different abstraction sets contain a varying number of elements A. Consequently, during training, we randomly select a subset of n elements, where $0 \leq n \leq N$ and N is the minimum between A and the maximum number of elements that fit into memory. As a result, the input parts embeddings cover only a fraction $\ominus = \frac{n}{A}$ of of the complete set. We employ DiT's modulation space to incorporate this coverage score into the velocity prediction v_θ. The continuous value $\ominus \in [0, 1]$ is encoded into Fourier features [48], followed by an MLP that projects these features to the modulation dimension. As seen in Fig. 3b (bottom-right), these score residuals affect only the image embedding modulations.

4.3 Enhanced Abstraction Control

During inference, the user provides an abstraction set (Fig. 3c, left). Changing the abstraction type of individual elements alters their generative role: (i) Aggressive augmentation (green outline) grants the model freedom to interpret and combine elements loosely (e.g., transforming the blue shape into elephant wheels). (ii) Mid-level augmentation (yellow outline) instructs the model to strictly preserve the shape within the composition. (iii) Soft augmentation (red outline) is the most restrictive; the model must maintain the element's exact position and generate the surrounding image (third elephant). (iv) Finally, inter-relation specifications (blue outline) define the spatial relationships between elements in the generated output.

Flow Less! To control overall adherence to the input abstraction set, the user can modify the coverage score $\ominus$. As illustrated in Figures 3c (right) and 4,

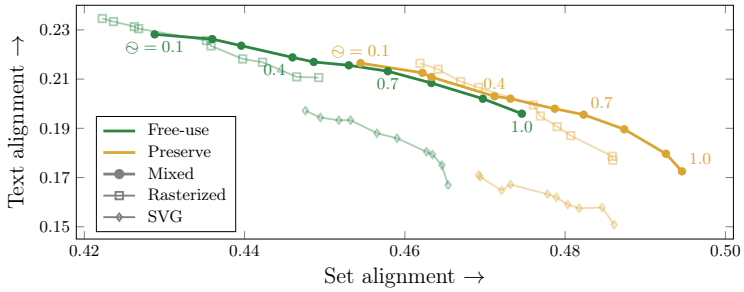


Fig. 5: Quantitative analysis. We compare the results of our method when varying coverage scores, dataset mixes, and abstraction set condition type.

increasing ≀, increases the abstraction level of the generated image, as the **flow** model implicitly learns to use **less** elements outside the provided abstraction set.

5 Experiments

In this section, we demonstrate the effectiveness of our framework through both qualitative and quantitative evaluations across diverse abstraction sets. We begin by detailing our implementation and data preparation pipeline, followed by a comprehensive performance analysis. We further compare FlowLess against state-of-the-art baselines and showcase its utility in creative tasks such as per-part control, compositional control via inter-relations, and image completion.

5.1 Implementation Details

Rasterized image decomposition. Our decomposition pipeline for rasterized images proceeds as follows: (i) We extract raw segments from each image by querying the *SegmentAnything* [40] model on a 128×128 grid followed by non-maxima suppression. (ii) We construct the relation graph as described in Sec. 4.1. (iii) We inpaint the occluded regions of each segment by utilizing the relation graph to define the inpainting mask. Specifically, for segment for segment i, we use the region of segment j as inpainting mask if it relates to i via intersection or inclusion.

Inter-relation extraction. Our detail elements are obtained by rendering a circular crop of a fixed radius for each edge in our relation graph, where the crop center is sampled along the intersection boundary between the two elements.

Datasets. Our final dataset is composed of 100K SVG illustrations and 50K synthetic rasterized illustrations. The SVGs were filtered from the MMSVG-Illustration dataset [61] to include at least three paths. The rasterized illustrations generated using Flux.1–dev from captions generated by Gemini [49].

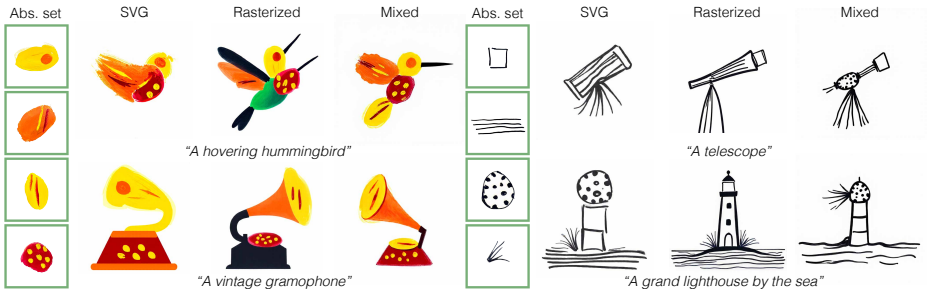


Fig. 6: Training dataset ablation. A FLowless model trained exclusively on SVG data yields lower-quality images (left columns).Training solely on rasterized data improves text adherence but limits the model's ability to produce strictly minimalist compositions (middle results). Our final model (right), trained on a mixed dataset, demonstrates a superior tradeoff by achieving high image quality while successfully constraining generation to the input primitives.

Training. We fine-tune the Flux.1–dev flow model for 4K iterations with a learning rate of 10^{-4}, using a batch size of 512 image–abstraction set pairs where each set include up to 8 elements. To increase weighting at high timesteps (more noise), we use a cosine scheduler with log-SNR shift of $2\log(2)$ [18]. We train using 64 TPUs-v3 devices with data parallelization and memory-efficient partitioning [38] for both model weights and optimizer states.

5.2 Evaluations

We evaluate our method over 1,000 image samples, generated with various model configurations and baseline methods. This evaluation set was generated using 100 fixed text prompts and 10 abstraction sets per prompt. The prompts were generated automatically [49] and consist of simple items such as animals, objects, and landscapes. Our abstraction sets utilize diverse materials and styles, including watercolor shapes, marker doodles, ink strokes, vector graphic primitives and more. Each set consists of 5–8 elements.

Metrics. We measure text alignment of each generated image to its condition text prompt via the common CLIP score [36]. We measure alignment to the condition abstraction set via the Chamfer similarity in DINOv2 feature space [6, 34]:

$$\mathcal{S}_{\text{Chamfer}}(\mathcal{X}, \mathcal{Y}) = \frac{1}{|\mathcal{X}|} \sum_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} s(x, y) + \frac{1}{|\mathcal{Y}|} \sum_{y \in \mathcal{Y}} \max_{x \in \mathcal{X}} s(x, y),$$

Where $\mathcal{X}$ and $\mathcal{Y}$ are DINOv2 features extracted from the generated image and the elements of the input abstraction set, respectively; and $s(x, y) = \dfrac{x \cdot y}{\|x\|\|y\|}$. Intuitively, set alignment measures bidirectional similarity: *fidelity* compares each generated image feature to its closest counterpart in the abstraction set, while *coverage* measures the reverse, mapping abstraction to image features.

Table 1: Quantitative comparison. Automatic metrics evaluate text alignment (CLIP score) and set consistency (DINO embedding similarity) between the generated images and the input text prompt and abstraction set, respectively. For perceptual human evaluation (User pref.), we report the percentage of times users preferred our method over the corresponding baseline in a 1-vs-1 setup.

Method	Text alignment ↑	Set alignment ↑	User pref.
InstantStyle	0.185 ± 0.046	0.454 ± 0.064	78.1 %
OmniControl	$\mathbf{0.242 \pm 0.033}$	0.409 ± 0.061	88.3 %
DreamO	0.235 ± 0.033	0.419 ± 0.062	93.5 %
Qwen-Image-Edit	0.239 ± 0.029	0.419 ± 0.067	79.1 %
Flux.2–dev	0.239 ± 0.029	0.429 ± 0.062	93.7 %
Flux.2–dev+ MMPE	0.207 ± 0.043	0.467 ± 0.078	80.9 %
Ours (Free-use)	0.196 ± 0.045	0.475 ± 0.073	74.5 %
Ours (Preserve)	0.172 ± 0.046	$\mathbf{0.495 \pm 0.065}$	–

Impact of modulation residuals. To evaluate our modulation conditioning, we repeated the generation process for 1,000 images with coverage scores varying from 0.1 to 1.0, and adding either *preservation* or *free-use* modulation residuals to the abstraction set embeddings. As shown quantitatively in Fig. 5, varying the coverage score reveals a tradeoff between text and set alignment. Furthermore, adding the *preservation* residuals improves set fidelity but decreases text alignment compared to the *free-use* residuals. The qualitative examples in Fig. 4 demonstrate this tradeoff where higher coverage scores yield more minimal and abstract images, e.g., fewer poppies, more abstract dinosaurs and tulips. Finally, the *preservation* condition better maintains the geometry of the distinct elements within the abstraction sets.

Impact of dataset mixture. To isolate the impact of data modality, we evaluated models trained exclusively on either SVG or rasterized images. The diamond-marked curves in Fig. 5 reveal a sharp performance degradation for the SVG-only model, which we attribute to its limited exposure to diverse artistic styles and materials. Conversely, the model trained only on rasterized images shows a performance drop when the coverage score exceeds ∼40–50 (square-marked curves in Fig. 5). This is consistent with the element distribution shown in Fig. 2a; specifically, the rasterized model is exposed to fewer images with low numbers of elements. Consequently, its ability to abstract and generate meaningful images from only a few elements is limited. The qualitative results in Fig. 6 illustrate the same phenomenon: images generated by the SVG-only model are of lower quality, while the raster-only model is less aligned with the abstraction set. The mixed-data model achieves superior abstraction and image quality.

Comparisons. To the best of our knowledge, there are no existing methods designed specifically for image generation conditioned on a *set of disjoint, non-semantic visual primitives*. Therefore, we compare our framework to state-of-the-art text-to-image open source baselines capable of conditioning generation

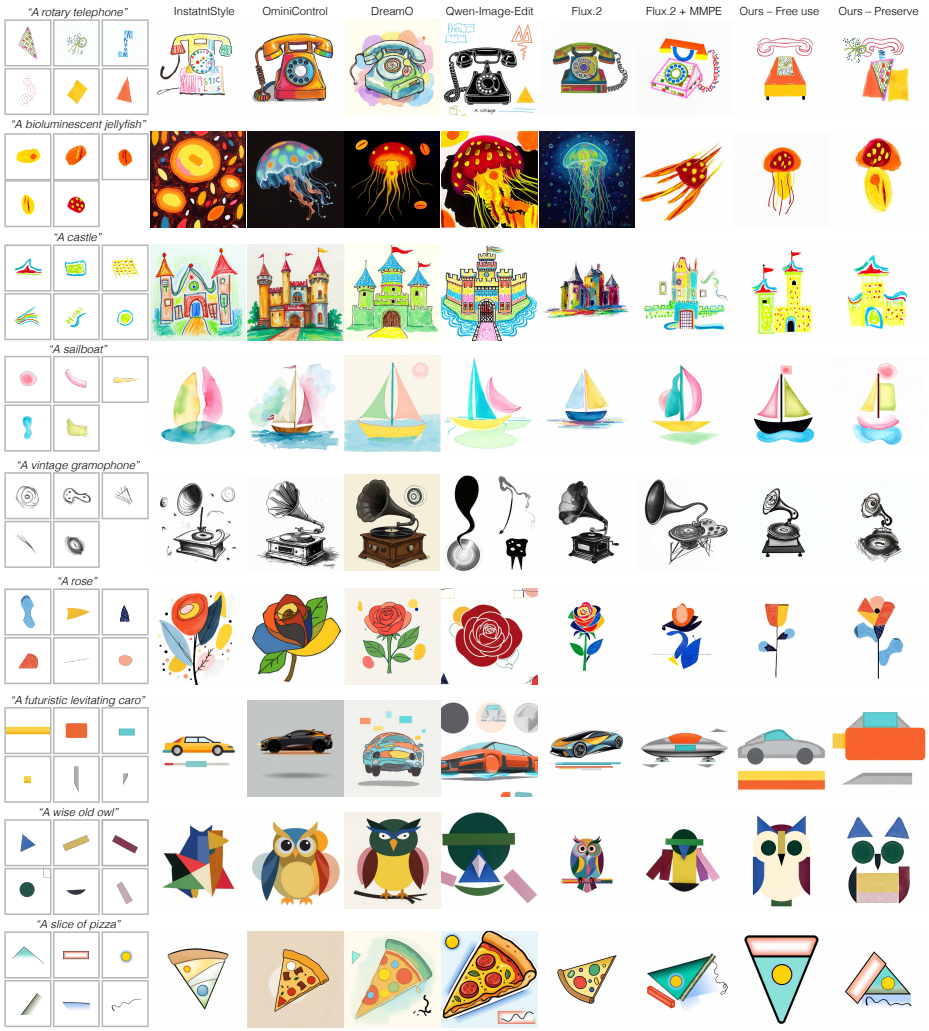


Fig. 7: Qualitative comparisons. Baseline methods generate high-quality images that adhere to the text prompt and capture the global style of the input abstraction set (left). Only our method (right columns) provides explicit abstraction control, yielding minimalist compositions that strictly match the textures (free use) and geometry (preserve) of the individual input elements under both.

on reference images. InstantStyle [54], DreamO [30], and OmniControl [47] adapt text-to-image flow models for customized image generation tasks, conditioning the Flux.1–dev model on style and subject reference images. Additionally, we compare our approach to Qwen-Image-Edit [58] and Flux.2–dev [24], which are general-purpose, context-aware flow models trained on a wide range of reference-

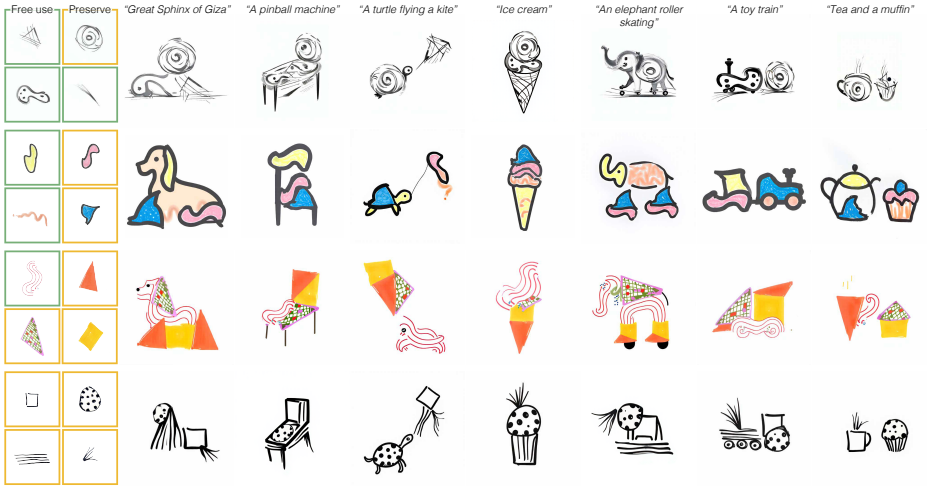


Fig. 8: Per part control. Our method enables per part abstraction control. On top, we specify the preservation of a single shape (yellow outline) while other elements (green outline) are combined freely into the generated image. On bottom we set high fidelity with respect to all four elements.

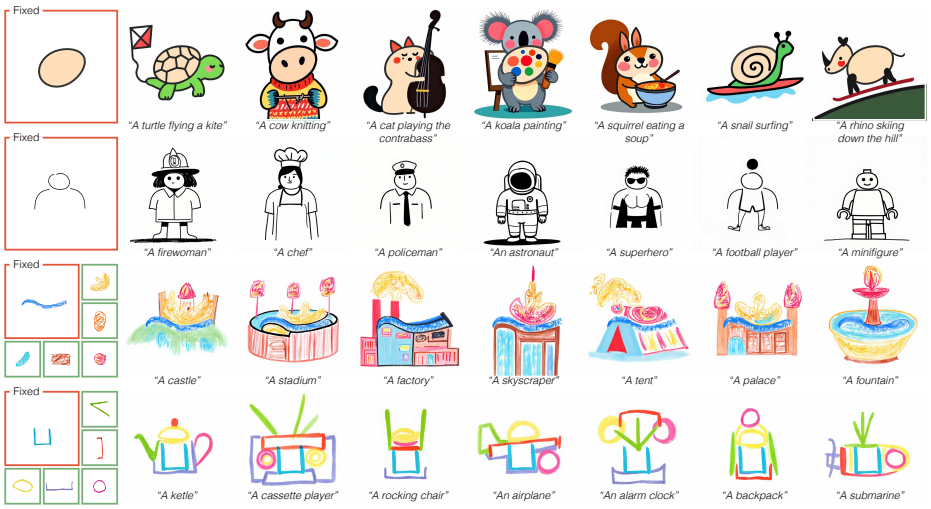
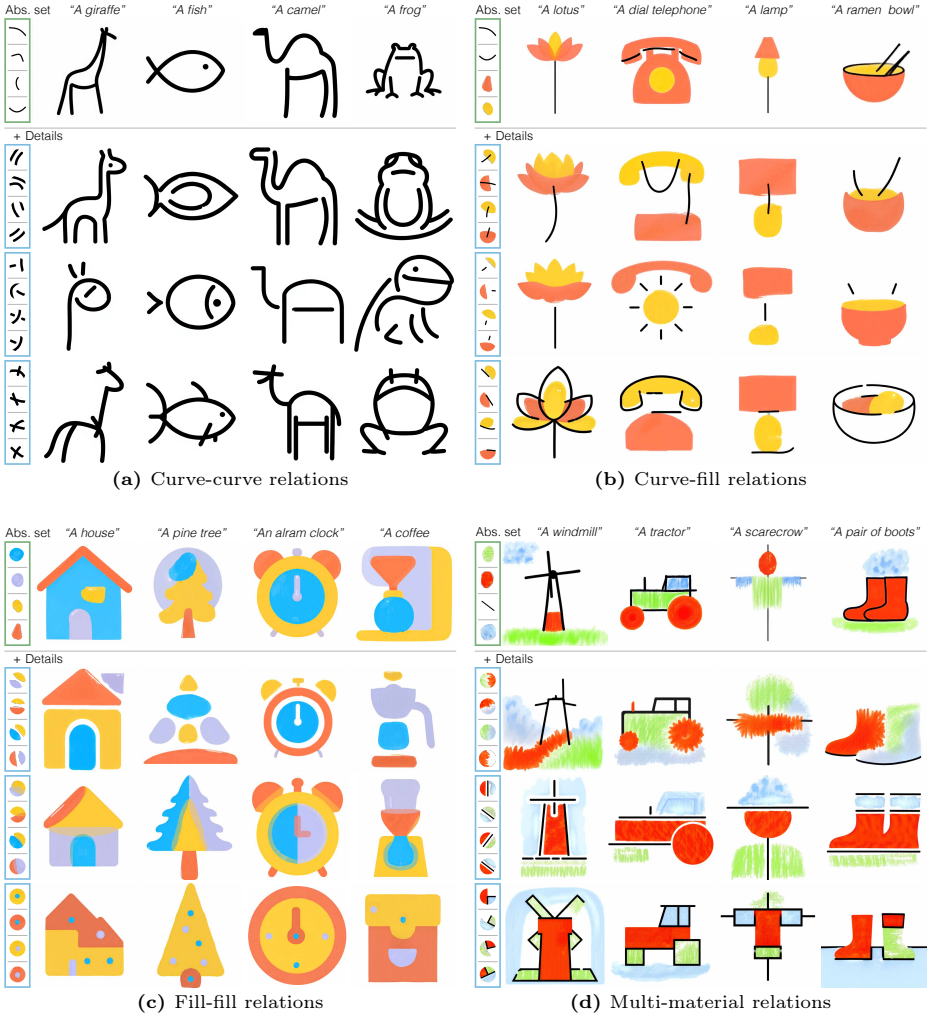


Fig. 9: Image completion. Our method can complete input drawings (red boxes) to generate different concepts using additional (optional) abstract elements (bottom).

based image generation tasks. Flux.2–dev optionally utilizes a Multimodal Prompt Extension (MMPE) model that provides precise descriptions of how each reference image should contribute to the generated output. Quantitative results are summarized in Table 1, and qualitative results are presented in Fig. 7. As shown, images generated by these baselines capture the global style of the input



(a) Curve-curve relations **(b)** Curve-fill relations

(c) Fill-fill relations **(d)** Multi-material relations

Fig. 10: Control by details. Top row in each group shows generation results using the abstraction elements in green outlines. In bottom rows: providing visual details (blue outlines) enables further abstraction control over the interrelations within the images.

set (e.g., marker strokes or pastel textures). However, they struggle to produce compositions strictly limited to the input abstraction set. In contrast, FlowLess enables granular control over the alignment with the individual elements of the abstraction set.

User study. To complement the automatic metrics, we conducted a perceptual user study with 20 participants (yielding 480 responses) in a two-alternative forced-choice setup. For each question, users were shown an input abstraction

set and a text prompt, followed by two generated images (our method vs. a baseline). They were then asked to select "which image better represents an abstract visualization of the desired content using the provided visual elements." The user preference scores for FlowLess compared to the baselines are reported in the final column of Table 1. Our method is consistently preferred over all other baselines, further validating the improved abstraction control achieved by our approach.

5.3 Additional results

Per-part control. As shown in Fig. 8, our method allows the user to specify the level of adherence required for individual elements within the abstraction set. For example, in the top row, we provide a set of ink elements and specify the preservation of the circular shape across different generations. By increasing the number of elements to *preserve* (bottom rows in Fig. 8), the model compose the input elements while better maintaining their original geometry.

Inter-relations control. In many cases, the appearance of an image is better realized via the relationships between elements than by the individual parts themselves. By providing visual details as part of the input abstraction set, users can specify these inter-relations and further control the final composition of the generated image. As shown in Fig. 10, adding different visual details to a fixed set of base elements enables further creative extrapolation as users can define specific inter-relations between curves, shapes, or materials to influence the overall look of the image.

Completion. Another creative use case of our method involves applying the *fixed* modulation residual , allowing the model to complete the image around it. Examples of this completion capability are shown in Fig. 9, where we use various pastel elements to illustrate different buildings constructed around a fixed blue wavy shape, complete different animals around an egg-like shape, or generate characters and objects emerging from the same initial strokes.

6 Conclusions

We have introduced FlowLess, a self-supervised framework for granular image control using a *visual abstraction set*. Our method enables the construction of diverse semantic concepts from disjoint visual primitives, offering precise control over abstraction levels, composition, and image completion. While *FlowLess* provides a powerful new paradigm for creative design, it is not *flawless*; it may struggle to synthesize concepts that deviate significantly from the provided abstraction set or fail to compose complex arrangements.

Looking ahead, we aim to integrate our framework with context-aware, multimodal generative models to enhance semantic reasoning. Another interesting and challenging future direction would be exploring abstraction control beyond static images into the temporal video domain—where one might extract and non-semantic motion styles, animation patterns, or dynamic temporal effects.

References

1. Alaluf, Y., Garibi, D., Patashnik, O., Averbuch-Elor, H., Cohen-Or, D.: Cross-image attention for zero-shot appearance transfer. In: ACM SIGGRAPH 2024 conference papers. pp. 1–12 (2024)
2. Alberti, L.B.: De re aedificatoria (On the Art of Building) (1485)
3. Avrahami, O., Aberman, K., Fried, O., Cohen-Or, D., Lischinski, D.: Break-a-scene: Extracting multiple concepts from a single image. In: SIGGRAPH Asia 2023 Conference Papers (2023). https://doi.org/10.1145/3610548.3618154
4. Avrahami, O., Lischinski, D., Fried, O.: Blended diffusion for text-driven editing of natural images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 18208–18218 (June 2022)
5. Byars, J.L.: UROBOROS: James Lee Byars encounters Leon Battista Alberti. Exhibition curated by Alberto Salvadori at Museo Marino Marini (May 30–November 8 2014)
6. Dall'Asen, N., Zhang, X., Askari-Hemmat, R., Hall, M., Verbeek, J., Romero-Soriano, A., Drozdzal, M.: Increasing the utility of synthetic images through chamfer guidance. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems (2025), https://openreview.net/forum?id=X3RVQNOIXZ
7. DeepMind, G.: Introducing nana banana pro (2025), https://blog.google/innovation-and-ai/products/nano-banana-pro/
8. Ding, G., Zhao, C., Wang, W., Yang, Z., Liu, Z., Chen, H., Shen, C.: Freecustom: Tuning-free customized image generation for multi-concept composition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2024)
9. Dorfman, S., Cohen-Bar, D., Gal, R., Cohen-Or, D.: Ip-composer: Semantic composition of visual concepts. In: Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers. pp. 1–11 (2025)
10. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2021), https://openreview.net/forum?id=YicbFdNTTy
11. Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., Podell, D., Dockhorn, T., English, Z., Rombach, R.: Scaling rectified flow transformers for high-resolution image synthesis. In: Proceedings of the 41st International Conference on Machine Learning. ICML'24, JMLR.org (2024)
12. Frans, K., Soros, L., Witkowski, O.: Clipdraw: Exploring text-to-drawing synthesis through language-image encoders. Advances in Neural Information Processing Systems **35**, 5207–5218 (2022)
13. Garibi, D., Yadin, S., Paiss, R., Tov, O., Zada, S., Ephrat, A., Michaeli, T., Mosseri, I., Dekel, T.: Tokenverse: Versatile multi-concept personalization in token modulation space. ACM Trans. Graph. **44**(4) (Jul 2025). https://doi.org/10.1145/3730843, https://doi.org/10.1145/3730843
14. Geng, D., Park, I., Owens, A.: Visual anagrams: Generating multi-view optical illusions with diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 24154–24163 (June 2024)

15. Goren, N., Yehezkel, S., Dahary, O., Voynov, A., Patashnik, O., Cohen-Or, D.: Visual diffusion models are geometric solvers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2026)
16. Greenberg, C.: Modernist painting. Arts Yearbook **4**, 103–108 (1961)
17. Hertz, A., Voynov, A., Fruchter, S., Cohen-Or, D.: Style Aligned Image Generation via Shared Attention . In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4775–4785. IEEE Computer Society, Los Alamitos, CA, USA (Jun 2024). https://doi.org/10.1109/CVPR52733.2024.00457, https://doi.ieeecomputersociety.org/10.1109/CVPR52733.2024.00457
18. Hoogeboom, E., Heek, J., Salimans, T.: Simple diffusion: end-to-end diffusion for high resolution images. In: Proceedings of the 40th International Conference on Machine Learning. ICML'23, JMLR.org (2023)
19. Hu, E.J., yelong shen, Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: LoRA: Low-rank adaptation of large language models. In: International Conference on Learning Representations (2022), https://openreview.net/forum?id=nZeVKeeFYf9
20. Iluz, S., Vinker, Y., Hertz, A., Berio, D., Cohen-Or, D., Shamir, A.: Word-as-image for semantic typography. ACM Transactions on Graphics (TOG) **42**(4), 1–11 (2023)
21. Jain, A., Xie, A., Abbeel, P.: Vectorfusion: Text-to-svg by abstracting pixel-based diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1911–1920 (2023)
22. Kumari, N., Zhang, B., Zhang, R., Shechtman, E., Zhu, J.Y.: Multi-concept customization of text-to-image diffusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2023)
23. Labs, B.F.: Flux. https://github.com/black-forest-labs/flux (2024)
24. Labs, B.F.: FLUX.2: Frontier Visual Intelligence. https://bfl.ai/blog/flux-2 (2025)
25. Li, T.M., Lukáč, M., Michaël, G., Ragan-Kelley, J.: Differentiable vector graphics rasterization for editing and learning. ACM Trans. Graph. (Proc. SIGGRAPH Asia) **39**(6), 193:1–193:15 (2020)
26. Lipman, Y., Chen, R.T.Q., Ben-Hamu, H., Nickel, M., Le, M.: Flow matching for generative modeling. In: The Eleventh International Conference on Learning Representations (2023), https://openreview.net/forum?id=PqvMRDCJT9t
27. Liu, X., Gong, C., Liu, Q.: Flow straight and fast: Learning to generate and transfer data with rectified flow. In: The Eleventh International Conference on Learning Representations (ICLR) (2023)
28. Loos, A.: Ornament and Crime. In: Opel, A. (ed.) Ornament and Crime: Selected Essays, pp. 167–176. Ariadne Press, Riverside, CA (1998), originally published as "Ornament und Verbrechen" in 1908
29. Magar, N., Hertz, A., Tabellion, E., Pritch, Y., Rav-Acha, A., Shamir, A., Hoshen, Y.: Lightlab: Controlling light sources in images with diffusion models. In: Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers. pp. 1–11 (2025)
30. Mou, C., Wu, Y., Wu, W., Guo, Z., Zhang, P., Cheng, Y., Luo, Y., Ding, F., Zhang, S., Li, X., Li, M., Liu, M., Jiang, Y., Wu, S., Zhao, S., Zhang, J., He, Q., Wu, X.: Dreamo: A unified framework for image customization. In: SIGGRAPH Asia 2025 Conference Papers (2025)
31. Ng, K.W., Zhu, X., Song, Y.Z., Xiang, T.: Partcraft: Crafting creative objects by parts. In: European Conference on Computer Vision (ECCV) (2024)

32. Nichol, A., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., McGrew, B., Sutskever, I., Chen, M.: GLIDE: Towards photorealistic image generation and editing with text-guided diffusion models. In: International Conference on Machine Learning (ICML). pp. 16784–16804. PMLR (2022)
33. OpenAI: Introducing gpt-4o image generation (2024), https://openai.com/index/introducing-4o-image-generation/
34. Oquab, M., Darcet, T., Moutakanni, T., Vo, H.V., Szafraniec, M., Khalidov, V., Fernandez, P., HAZIZA, D., Massa, F., El-Nouby, A., Assran, M., Ballas, N., Galuba, W., Howes, R., Huang, P.Y., Li, S.W., Misra, I., Rabbat, M., Sharma, V., Synnaeve, G., Xu, H., Jegou, H., Mairal, J., Labatut, P., Joulin, A., Bojanowski, P.: DINOv2: Learning robust visual features without supervision. Transactions on Machine Learning Research (2024), https://openreview.net/forum?id=a68SUt6zFt, featured Certification
35. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 4195–4205 (2023)
36. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision. In: Meila, M., Zhang, T. (eds.) Proceedings of the 38th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 139, pp. 8748–8763. PMLR (18–24 Jul 2021), https://proceedings.mlr.press/v139/radford21a.html
37. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. J. Mach. Learn. Res. **21**(1) (Jan 2020)
38. Rajbhandari, S., Rasley, J., Ruwase, O., He, Y.: ZeRO: Memory optimizations toward training trillion parameter models. In: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC). IEEE (2020)
39. Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., Chen, M.: Hierarchical text-conditional image generation with clip latents. arXiv preprint arXiv:2204.06125 (2022)
40. Ravi, N., Gabeur, V., Hu, Y.T., Hu, R., Ryali, C., Ma, T., Khedr, H., Rädle, R., Rolland, C., Gustafson, L., Mintun, E., Pan, J., Alwala, K.V., Carion, N., Wu, C.Y., Girshick, R., Dollár, P., Feichtenhofer, C.: SAM 2: Segment anything in images and videos. In: The Thirteenth International Conference on Learning Representations (ICLR) (2025)
41. Richardson, E., Goldberg, K., Alaluf, Y., Cohen-Or, D.: Piece it together: Part-based concepting with ip-priors. arXiv preprint arXiv:2503.10365 (2025)
42. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: 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)
43. Rose, B.: ABC Art. Art in America **53**(5), 57–69 (October 1965), seminal essay on Minimalism
44. Rout, L., Chen, Y., Ruiz, N., Kumar, A., Caramanis, C., Shakkottai, S., Chu, W.S.: Rb-modulation: Training-free stylization using reference-based modulation. In: The Thirteenth International Conference on Learning Representations (2025)
45. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E., Ghasemipour, S.K.S., Ayan, B.K., Mahdavi, S.S., Lopes, R.G., Salimans, T., Ho, J., Fleet, D.J., Norouzi, M.: Photorealistic text-to-image diffusion models with deep language understanding. In: Advances in Neural Information Processing Systems (NeurIPS) (2022)

46. Sohn, K., Jiang, L., Barber, J., Lee, K., Ruiz, N., Krishnan, D., Chang, H., Li, Y., Essa, I., Rubinstein, M., Hao, Y., Entis, G., Blok, I., Chin, D.C.: Styledrop: Text-to-image synthesis of any style. In: Thirty-seventh Conference on Neural Information Processing Systems (2023), https://openreview.net/forum?id=KoaFh16uOc
47. Tan, Z., Liu, S., Yang, X., Xue, Q., Wang, X.: Ominicontrol: Minimal and universal control for diffusion transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 14940–14950 (2025)
48. Tancik, M., Srinivasan, P.P., Mildenhall, B., Fridovich-Keil, S., Raghavan, N., Singhal, U., Ramamoorthi, R., Barron, J.T., Ng, R.: Fourier features let networks learn high frequency functions in low dimensional domains. NeurIPS (2020)
49. Team, G., Anil, R., Borgeaud, S., Alayrac, J.B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A.M., Hauth, A., Millican, K., et al.: Gemini: a family of highly capable multimodal models. arXiv preprint arXiv:2312.11805 (2023)
50. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I.: Attention is all you need. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 30. Curran Associates, Inc. (2017), https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf
51. Vinker, Y., Voynov, A., Tagliasacchi, A., Patrashin, O., Cohen-Or, D., Bermano, A.H.: Concept decomposition for visual exploration and inspiration. ACM Transactions on Graphics (TOG) **42**(6), 1–15 (2023)
52. Voynov, A., Aberman, K., Cohen-Or, D.: Sketch-guided text-to-image diffusion models. In: ACM SIGGRAPH 2023 conference proceedings. pp. 1–11 (2023)
53. Wan, T., Wang, A., Ai, B., Wen, B., Mao, C., Xie, C.W., Chen, D., Yu, F., Zhao, H., Yang, J., Zeng, J., Wang, J., Zhang, J., Zhou, J., Wang, J., Chen, J., Zhu, K., Zhao, K., Yan, K., Huang, L., Feng, M., Zhang, N., Li, P., Wu, P., Chu, R., Feng, R., Zhang, S., Sun, S., Fang, T., Wang, T., Gui, T., Weng, T., Shen, T., Lin, W., Wang, W., Wang, W., Zhou, W., Wang, W., Shen, W., Yu, W., Shi, X., Huang, X., Xu, X., Kou, Y., Lv, Y., Li, Y., Liu, Y., Wang, Y., Zhang, Y., Huang, Y., Li, Y., Wu, Y., Liu, Y., Pan, Y., Zheng, Y., Hong, Y., Shi, Y., Feng, Y., Jiang, Z., Han, Z., Wu, Z.F., Liu, Z.: Wan: Open and advanced large-scale video generative models. arXiv preprint arXiv:2503.20314 (2025)
54. Wang, H., Wang, Q., Bai, X., Qin, Z., Chen, A.: Instantstyle: Free lunch towards style-preserving in text-to-image generation. arXiv preprint arXiv:2404.02733 (2024)
55. Wang, S., Saharia, C., Montgomery, C., Pont-Tuset, J., Noy, S., Pellegrini, S., Onoe, Y., Laszlo, S., Fleet, D.J., Soricut, R., et al.: Imagen editor and editbench: Advancing and evaluating text-guided image inpainting. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 18359–18369 (2023)
56. Wang, X., Kontkanen, J., Curless, B., Seitz, S.M., Kemelmacher-Shlizerman, I., Mildenhall, B., Srinivasan, P., Verbin, D., Holynski, A.: Generative powers of ten. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 7173–7182 (June 2024)
57. Wang, Y., Lian, Z.: Deepvecfont: synthesizing high-quality vector fonts via dual-modality learning. ACM Transactions on Graphics (TOG) **40**(6), 1–15 (2021)
58. Wu, C., Li, J., Zhou, J., Lin, J., et al.: Qwen-image technical report. arXiv preprint arXiv:2508.02324 (2025)

59. Wu, R., Su, W., Ma, K., Liao, J.: Iconshop: Text-guided vector icon synthesis with autoregressive transformers. ACM Transactions on Graphics (TOG) **42**(6), 1–14 (2023)
60. Xiao, S., Wang, Y., Zhou, J., Yuan, H., Xing, X., Yan, R., Li, C., Wang, S., Huang, T., Liu, Z.: Omnigen: Unified image generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 13294–13304 (2025)
61. Yang, Y., Cheng, W., Chen, S., Zeng, X., Yin, F., Zhang, J., Wang, L., YU, G., Ma, X., Jiang, Y.G.: OmniSVG: A unified scalable vector graphics generation model. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems (2025), https://openreview.net/forum?id=sRLB4XjorC
62. Ye, H., Zhang, J., Liu, S., Han, X., Yang, W.: Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. arXiv preprint arXiv:2308.06721 (2023)
63. Zhang, G., Sohn, K., Hahn, M., Shi, H., Essa, I.: Finestyle: Fine-grained controllable style personalization for text-to-image models. In: Globerson, A., Mackey, L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J., Zhang, C. (eds.) Advances in Neural Information Processing Systems. vol. 37, pp. 52937–52961. Curran Associates, Inc. (2024). https://doi.org/10.52202/079017-1677, https://proceedings.neurips.cc/paper_files/paper/2024/file/5edb0d52b6959d946afac7600f9f1e0c-Paper-Conference.pdf
64. Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models (2023)
65. Zhou, Y., Gao, X., Chen, Z., Huang, H.: Attention distillation: A unified approach to visual characteristics transfer. In: 2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 18270–18280 (2025). https://doi.org/10.1109/CVPR52734.2025.01703