KINGFISHER: FAST PERSONALIZED VISUAL AUTOREGRESSIVE GENERATION VIA SCALE-AWARE PROMPT-CONDITIONED SCALE DECOMPOSITION

Anonymous authors
Paper under double-blind review

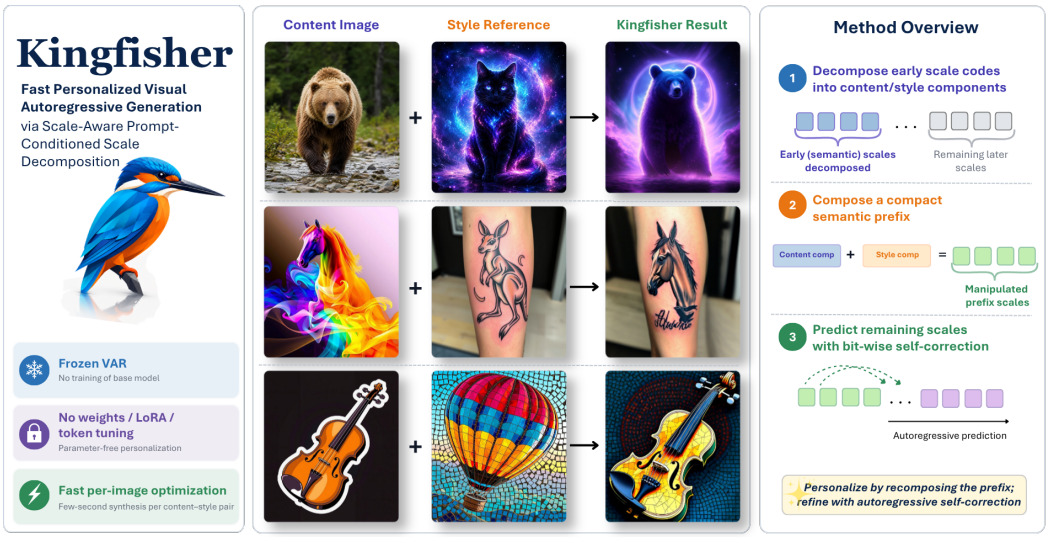

Figure 1: **Kingfisher** combines the content of one real image with the appearance of another by decomposing and recomposing early autoregressive scale codes. The remaining scales are predicted by the frozen VAR generator via bit-wise self-correction, enabling fast, coherent personalization without model tuning, LoRA, adapters, or learned subject embeddings.

ABSTRACT

Personalized image generation has been extensively studied in diffusion models, often relying on subject-specific optimization of model weights, textual embeddings, LoRA modules, or adapters. In this work, we explore a different direction for fast personalization in visual autoregressive models. Our key observation is that a short prefix of early autoregressive scale codes is sufficient to strongly determine both object structure and global appearance, while later scales primarily refine the established generation trajectory. Based on this insight, we introduce Kingfisher, a model-tuning-free framework that decomposes early residual scale codes into content- and style-related components through prompt-conditioned scale-code subspaces. Given two real images, Kingfisher first decomposes their early scale codes, then recomposes the content components of one image with the style components of the other into a compact semantic prefix. The frozen generator subsequently predicts the remaining scales through bitwise self-correction for coherent refinement. Experiments on CSD-100 show that Kingfisher achieves strong content preservation and the highest style and text alignment among the evaluated methods, while reducing pair-specific preparation time by approximately $54\times$ compared with diffusion-based methods and $7.5\times$ compared with VAR-based alternatives. These results demonstrate that effective personalization can be achieved by directly manipulating the native scale-space representation of a frozen visual autoregressive generator.

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

1 INTRODUCTION

Personalized image generation aims to adapt a generative model to user-provided visual concepts, enabling the synthesis of customized subjects, styles, or compositions under new conditions. Recent surveys Wei et al. (2025); Xu et al. (2025) organize this problem across different generative architectures through inversion spaces Ho et al. (2020), inversion methods Song et al. (2020), and personalization schemes, covering GAN-based methods Richardson et al. (2021); Abdal et al. (2019), diffusion personalization, and emerging autoregressive or multimodal generators. In diffusion models, existing methods personalize generation by operating in different representation spaces, including textual spaces Gal et al. (2022); Ruiz et al. (2023), feature spaces Wei et al. (2023); Shi et al. (2024), noise spaces, and parameter spaces Hu et al. (2022); Shah et al. (2024). These methods are further realized through training-free inversion Mokady et al. (2023), optimization-based concept learning Kumari et al. (2023), learning-based encoders or adapters Shi et al. (2024); Ye et al. (2023); Ma et al. (2024), and hybrid personalization pipelines Wang et al. (2024); Li et al. (2024). Beyond single-subject customization, recent work also explores style-driven personalization and visual concept composition Vinker et al. (2023); Frenkel et al. (2024); Dorfman et al. (2025); Jin et al. (2024); Wang et al. (2025); Gu et al. (2023).

In parallel, visual autoregressive models have emerged as an alternative paradigm for image synthesis. In contrast to conventional token-by-token generation, Visual Autoregressive Modeling (VAR) Tian et al. (2024) formulates generation as coarse-to-fine next-scale prediction, while Infinity Han et al. (2025) further introduces bitwise autoregressive modeling and self-correction for high-resolution text-to-image synthesis. This explicit scale-wise generation process suggests a different route for personalization: rather than adapting model parameters or auxiliary representations, one can directly operate on the native scale-code trajectory of a frozen generator. However, how content and style are organized across the scale hierarchy of visual autoregressive models remains underexplored. Diffusion-based personalization methods commonly represent visual factors through learned textual tokens, LoRA parameters, or image-conditioned features Gal et al. (2022); Ruiz et al. (2023); Frenkel et al. (2024); Liu et al. (2025). Recent VAR-based works begin to reveal the importance of scale-wise representations: CSD-VAR Nguyen et al. (2025) studies scale-aware content-style decomposition, while training-free style personalization methods identify pivotal intermediate scales whose dominant components strongly affect style Lee et al. (2026). These works provide valuable evidence that individual scales encode different visual factors, but they largely analyze or intervene on scales independently. Motivated by this gap, through a series of controlled experiments, we examine consecutive scales as an autoregressive generation trajectory and observe that early scales act as semantic anchors that constrain subsequent visual refinements. This finding suggests that personalization need not modify the full generation sequence; composing a compact prefix is sufficient to steer content and appearance, while the frozen generator completes the remaining scales through autoregressive prediction.

We introduce **Kingfisher**, a model-tuning-free framework for fast personalized visual autoregressive generation. Motivated by the observation that early autoregressive scale codes are neither purely content nor purely style, but instead contain mixed content, appearance, and residual factors, Kingfisher models each scale code as an approximate composition $R_i \approx C_i + S_i + e_i$, where C_i captures object-related structure, S_i captures visual appearance, and e_i absorbs residual information. To estimate these components without introducing new trainable parameters, Kingfisher constructs prompt-conditioned scale-code subspaces from the frozen generator itself: neutral prompts induce a neutral-style reference space, while object-specific prompts induce content directions relative to this reference. The decomposition is obtained by encouraging the style-removed code $R_i - S_i$ to lie near the neutral-style subspace, the content-removed code $R_i - C_i$ to suppress object-related directions, and the estimated content and style components to remain weakly coupled. Kingfisher then recomposes content factors from one real image and style factors from another within a compact semantic prefix, while the frozen autoregressive generator predicts the remaining scales through bitwise self-correction for coherent refinement. Although this formulation can be extended to other prompt-definable visual factors by constructing corresponding prompt-conditioned subspaces, we focus in this work on content–style personalization. We evaluate Kingfisher on CSD-100 Nguyen et al. (2025) and show that, compared with prior diffusion-based personalization methods and VAR-based decomposition methods, it achieves a favorable trade-off between source-content preservation and reference-appearance adoption while substantially improving runtime efficiency. These results

demonstrate effective content–style personalization without optimizing model weights, LoRA modules, adapters, or subject-specific embeddings.

Our contributions are summarized as follows:

- We show that a short prefix of early autoregressive scale codes is sufficient to strongly control object structure and global appearance, revealing these scales as a compact semantic bottleneck for personalization.
- We propose prompt-conditioned scale-code decomposition, which constructs neutral-style and object-content subspaces to explicitly decompose early scale codes into content- and style-related components in a frozen generator.
- We introduce a scale-aware personalization pipeline that recomposes content and style components across two real images within a compact semantic prefix, while relying on autoregressive bitwise self-correction to generate the remaining scales for coherent refinement.
- We demonstrate that Kingfisher achieves the highest style and text alignment among the evaluated methods on CSD-100 while maintaining strong content preservation, with approximately $\mathbf{54}\times$ and $\mathbf{7.5}\times$ faster pair-specific preparation than diffusion- and VAR-based alternatives, respectively.

2 Related Work

Personalized image generation. Personalized image generation has been extensively studied in diffusion models, where user-provided concepts are represented through textual embeddings Gal et al. (2022), subject-specific fine-tuning Ruiz et al. (2023), parameter-efficient updates such as LoRA Hu et al. (2022); Shah et al. (2024); Frenkel et al. (2024), or image-conditioning adapters and encoders Ye et al. (2023); Wei et al. (2023); Ma et al. (2024); Shi et al. (2024). Recent methods further explore style-driven personalization, multi-concept composition, and concept disentanglement Vinker et al. (2023); Dorfman et al. (2025); Liu et al. (2025). Beyond diffusion personalization, prior works on GAN inversion and latent editing also show that visual factors can be organized across generator layers or latent spaces. StyleGAN Karras et al. (2019; 2020) provides scale-specific control through different synthesis layers, and StyleSpace further identifies localized channel-wise controls for distinct visual attributes Wu et al. (2021). Inversion methods extend the latent space to $\mathcal{W}+$ Tov et al. (2021) for more accurate reconstruction and editing of real images. Similar layer- or stage-wise control has been explored in diffusion models, where $P+$ Voynov et al. (2023) uses per-layer prompt embeddings for more disentangled control and object–style mixing. These works suggest that generator layers or stages can expose controllable visual factors. In contrast, Kingfisher studies this question in visual autoregressive models and shows that early residual scales form entangled semantic bottlenecks, where object structure, appearance, composition, and style priors are coupled rather than cleanly separated.

Visual autoregressive generation. Autoregressive image generation models visual synthesis as sequential prediction Van den Oord et al. (2016); Reed et al. (2017); Chen et al. (2020). Discrete image representation learning, such as VQ-VAE and VQGAN, provides quantized visual codes for compact image modeling Diederik & Max (2019); Van Den Oord et al. (2017); Razavi et al. (2019); Esser et al. (2021). When images are represented as discrete visual tokens, token-based autoregressive generators typically flatten them into one-dimensional code sequences and generate them in raster or predefined orders Esser et al. (2021); Yu et al. (2022). In contrast, Visual Autoregressive Modeling (VAR) reformulates generation as coarse-to-fine next-scale prediction Tian et al. (2024). Infinity further introduces bitwise visual autoregressive modeling and bitwise self-correction for high-resolution text-to-image synthesis Han et al. (2025). This explicit scale hierarchy motivates us to study how early scale codes can be decomposed and recomposed for fast personalization.

3 Scale-wise Insights and Preliminaries

Controlled scale-wise analysis. We analyze how visual information is organized across the residual scales of Infinity, where each image is represented by $N = 13$ residual scale codes $\{R_i\}_{i=1}^{N}$. As shown in Fig. 2, prefix reconstruction preserves the first k ground-truth scales and regenerates the remaining ones under an unconditional prompt; the first four scales already nearly reconstruct the input image. Prompt routing assigns Prompt_2 to selected scale ranges and Prompt_1 to the rest. Even when Prompt_2 controls large suffixes such as $R_{5:13}$ or $R_{10:13}$, the output still follows Prompt_1, and shifts toward Prompt_2 only when early scales such as $R_{1:3}$ or $R_{2:4}$ are routed to it. Appendix A

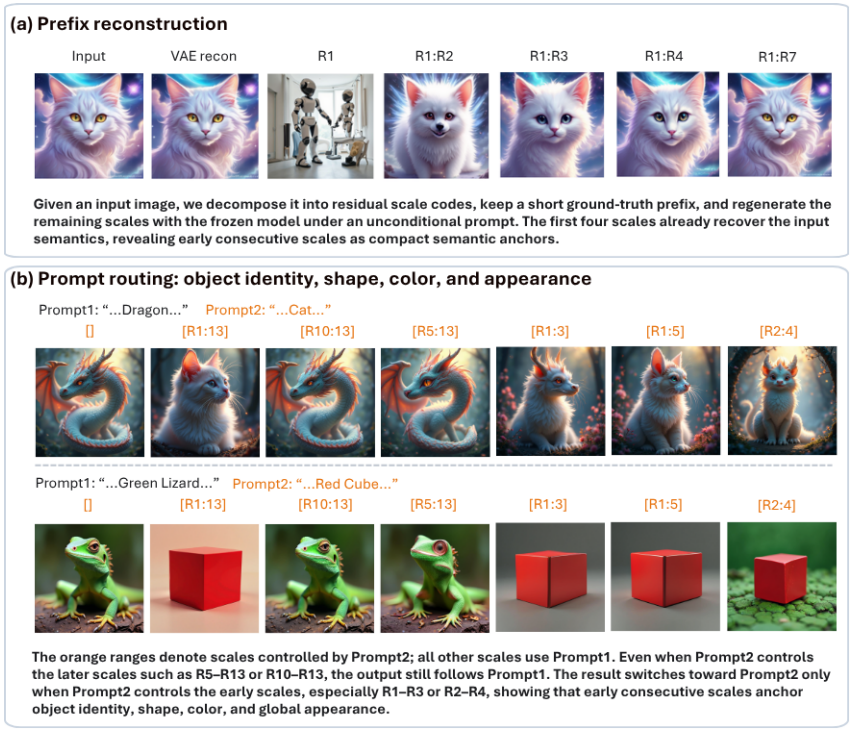

Figure 2: **Scale-wise analysis of visual autoregressive generation.** We analyze residual scales from two complementary views: prefix reconstruction and prompt routing. Prefix reconstruction shows that a short early ground-truth prefix is nearly sufficient to recover input semantics, while prompt routing shows that early scales dominate object identity, shape, color, and global appearance. Together, these results indicate that early consecutive scales form compact yet entangled semantic anchors, whereas later scales mainly refine details.

reports quantitative results for the prefix-reconstruction and prompt-routing experiments across different residual scale ranges. These results suggest that early residual scales act as semantic anchors, while later scales primarily refine local visual details.

Visual autoregressive generation and implication. VAR formulates image synthesis as coarse-to-fine next-scale prediction rather than raster-ordered token generation Tian et al. (2024).We denote the residual scale codes by $\{R_i\}_{i=1}^{N}$, where N is the total number of scales. For compact notation, let $R_{a:b} \triangleq (R_a, \ldots, R_b)$ and $R_{<i} \triangleq (R_1, \ldots, R_{i-1})$. The autoregressive generation process can then be factorized as:

$$p(R_{1:N} \mid y) = \prod_{i=1}^{N} p(R_i \mid R_{<i}, y), \tag{1}$$

where y is the text prompt. Infinity Han et al. (2025) follows this scale-wise autoregressive formulation with bitwise visual token prediction and self-correction. At step $i+1$, the model predicts R_{i+1} from the accumulated visual prefix:

$$Z_i = \sum_{j=1}^{i} \mathrm{Up}(R_j), \tag{2}$$

where $\mathrm{Up}(\cdot)$ interpolates each residual scale to the maximum spatial resolution $H \times W$ before aggregation. The prediction also attends autoregressively to the previously computed prefix states $\{Z_j\}_{j=1}^{i}$. Thus, predicting R_5 depends on Z_4, accumulated from $R_{1:4}$, as well as earlier prefix states $Z_{1:3}$. This repeated conditioning helps explain why early residual scales act as semantic anchors: later scales mostly refine details within the trajectory established by the early prefix. Therefore, the residual scale hierarchy is not a clean content-style separation; early consecutive scales instead form an entangled semantic bottleneck coupling object identity, shape, color, background, lighting, and style priors. This motivates our Kingfisher to decompose and recompose content and appearance

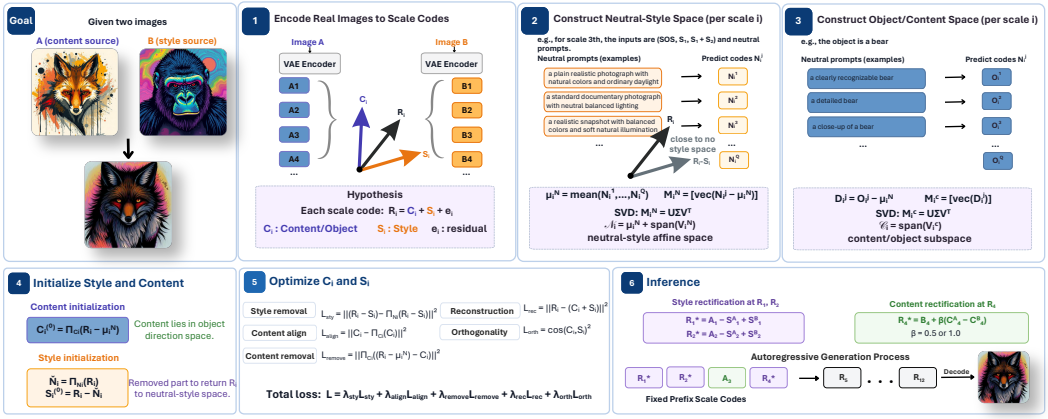

Figure 3: **Overview of the Kingfisher pipeline.** Given a content image A and a style image B, both are encoded into multi-scale residual codes using the frozen tokenizer. Kingfisher constructs a neutral-style affine space from neutral prompts and an object-content subspace from object-specific prompts. The selected early residual codes are decomposed into content, style, and residual factors through prompt-space constraints and recomposed into a compact semantic prefix. The frozen Infinity generator then predicts the remaining scales through bitwise self-correction, producing an output that preserves the object structure of A while adopting the visual appearance of B.

factors only within the compact early prefix, while leaving the frozen generator to complete later scales through bitwise self-correction.

4 Proposed Method

Overview. As illustrated in Fig. 3, given a content image I^A and a style reference I^B, Kingfisher generates an image that preserves the object semantics and structure of I^A while adopting the visual appearance of I^B. The frozen Infinity tokenizer encodes the two images into residual scale-code sequences $\{A_i\}_{i=1}^N$ and $\{B_i\}_{i=1}^N$, respectively. Guided by the scale-wise analysis in Sec. 3, Kingfisher operates only on the first four scales, where prompt-conditioned subspaces are used to estimate content and style components. These components are recomposed into a compact semantic prefix, while the frozen Infinity generator predicts the remaining scales through bitwise self-correction.

4.1 Prompt-Conditioned Scale Subspaces

For a real image encoded as $\{R_i\}_{i=1}^N$, where N is number of residual scales, we query the frozen generator at scale i-th under prompt p, while keeping the preceding scale-code prefix $R_{<i}$ fixed:

$$\widehat{R}_i(p) = \mathcal{G}_i(R_{<i}, p). \tag{3}$$

This isolates the influence of the prompt at scale i without changing its autoregressive image context.

Given neutral prompts $\mathcal{P}^N = \{p_q^N\}_{q=1}^{Q_N}$, where the superscript N denotes neutral and Q_N is the number of neutral prompts, we obtain neutral predictions $N_i^q = \widehat{R}_i(p_q^N)$. Their mean, centered matrix, and SVD are defined compactly as:

$$\mu_i^N = \frac{1}{Q_N}\sum_{q=1}^{Q_N} N_i^q, \qquad M_i^N = \left[\text{vec}(N_i^q - \mu_i^N)^\top\right]_{q=1}^{Q_N} = U_i^N \Sigma_i^N (V_i^N)^\top. \tag{4}$$

Using the first r_N right singular vectors, we construct the neutral-style affine subspace:

$$\mathcal{N}_i = \mu_i^N + \text{span}(V_{i,1:r_N}^N). \tag{5}$$

The neutral prompts describe natural colors, balanced illumination, and no explicit artistic stylization, providing an operational neutral-appearance reference.

Similarly, object prompts $\mathcal{P}^C = \{p_q^C\}_{q=1}^{Q_C}$ produce $O_i^q = \widehat{R}_i(p_q^C)$, where the superscript C denotes content and Q_C is number of content/object prompts. Since O_i^q is an absolute scale code, it contains both object-related information and shared appearance components. We therefore subtract the neutral-style mean μ_i^N to remove this common baseline and isolate the code variation induced by the object prompt:

$$D_i^q = O_i^q - \mu_i^N. \tag{6}$$

Each D_i^q can thus be interpreted as a direction from the neutral reference toward the target object. We stack these directions, apply SVD, and define the object-content subspace as:

$$M_i^C = \left[\mathrm{vec}(D_i^q)^\top\right]_{q=1}^{Q_C} = U_i^C \Sigma_i^C (V_i^C)^\top, \qquad \mathcal{C}_i = \mathrm{span}(V_{i,1:r_C}^C). \tag{7}$$

While $\mathcal{N}_i$ models variations around a neutral appearance reference, sub space $\mathcal{C}_i$ captures the principal directions associated with the target object.

4.2 Prompt-Conditioned Scale Decomposition

For each selected scale i, we decompose the vectorized scale code as $R_i = C_i + S_i + e_i$, where C_i, S_i, and e_i denote object-content, visual-appearance, and unexplained residual components, respectively.

Component initialization. Before joint optimization, we initialize C_i and S_i using the geometry of the two prompt-conditioned subspaces. Let $\bar{N}_i = \Pi_{\mathcal{N}_i}(R_i)$ be the closest neutral-style code to R_i. We initialize:

$$S_i^{(0)} = R_i - \bar{N}_i \quad \Longrightarrow \quad R_i - S_i^{(0)} = \bar{N}_i, \qquad C_i^{(0)} = \Pi_{\mathcal{C}_i}(R_i - \mu_i^N), \tag{8}$$

where $\Pi_{\mathcal{N}_i}(\cdot)$ and $\Pi_{\mathcal{C}_i}(\cdot)$ denote projections onto the neutral-style and object-content subspaces, respectively.

Thus, $S_i^{(0)}$ is exactly the component whose removal places R_i on the neutral-style subspace, while $C_i^{(0)}$ extracts the object-related variation after removing the neutral baseline. so removing the initial style component immediately yields a neutral-style code. For content, we first subtract the neutral mean to remove the shared appearance baseline, then project the remaining variation onto $\mathcal{C}_i$. Thus, $C_i^{(0)}$ is the object-aligned component of R_i that is best represented by the content subspace. These geometry-aware initializations provide meaningful starting points and reduce ambiguity during joint decomposition.

Starting from these initializations, we jointly optimize C_i and S_i using the following constraints:

$$\begin{aligned} \mathcal{L}_{\mathrm{sty}} &= \|(R_i - S_i) - \Pi_{\mathcal{N}_i}(R_i - S_i)\|_2^2, \qquad \mathcal{L}_{\mathrm{align}} = \|C_i - \Pi_{\mathcal{C}_i}(C_i)\|_2^2, \\ \mathcal{L}_{\mathrm{remove}} &= \left\|\Pi_{\mathcal{C}_i}((R_i - \mu_i^N) - C_i)\right\|_2^2, \qquad \mathcal{L}_{\mathrm{rec}} = \|R_i - (C_i + S_i)\|_2^2, \\ \mathcal{L}_{\mathrm{orth}} &= \cos^2(C_i, S_i). \end{aligned} \tag{9}$$

Here, $\mathcal{L}_{\mathrm{sty}}$ makes the style-removed code close to the neutral subspace, while $\mathcal{L}_{\mathrm{align}}$ constrains C_i to follow object-related directions. Since alignment alone does not ensure that C_i captures all content information, $\mathcal{L}_{\mathrm{remove}}$ suppresses any content-subspace component remaining after C_i is removed. Finally, $\mathcal{L}_{\mathrm{rec}}$ preserves the original information and $\mathcal{L}_{\mathrm{orth}}$ reduces overlap between content and appearance. The total objective is:

$$\mathcal{L} = \lambda_{\mathrm{sty}}\mathcal{L}_{\mathrm{sty}} + \lambda_{\mathrm{align}}\mathcal{L}_{\mathrm{align}} + \lambda_{\mathrm{remove}}\mathcal{L}_{\mathrm{remove}} + \lambda_{\mathrm{rec}}\mathcal{L}_{\mathrm{rec}} + \lambda_{\mathrm{orth}}\mathcal{L}_{\mathrm{orth}}. \tag{10}$$

4.3 Scale-Aware Prefix Composition

Given a content image A and a style image B, we obtain (C_i^A, S_i^A) and (C_i^B, S_i^B) from the decomposition and recombine them to construct the early semantic prefix. Our preliminary experiments with different early-scale compositions show that the best balance between content preservation and appearance transfer is achieved by rectifying the appearance components at the first two scales, preserving the third scale as a semantic anchor, and correcting the fourth scale using a content direction. Specifically, we define:

$$R_i^* = A_i - S_i^A + S_i^B, \quad i \in \{1, 2\}, \qquad R_3^* = A_3, \qquad R_4^* = B_4 + \beta(C_4^A - C_4^B), \tag{11}$$

where scale code A provides the target content, scale code B provides the reference appearance, and β controls the strength of the structural correction at R_4. The resulting semantic prefix is:

$$\mathcal{P}^* = [R_1^*, R_2^*, A_3, R_4^*]. \tag{12}$$

At R_1 and R_2, appearance rectification retains the original content code from A while replacing only its estimated appearance component with that of B. In contrast, A_3 is preserved directly because modifying this scale tends to disturb object identity and structure. Finally, R_4^* uses B_4 as an appearance-aware base and adds the content difference $C_4^A - C_4^B$ to steer it toward the structure of A. Thus, the prefix combines coarse appearance transfer, a preserved semantic anchor, and content-aware structural correction.

We do not copy the remaining scales from either image, since later real-image codes may reintroduce source-specific details. Instead, the frozen generator autoregressively predicts the suffix conditioned on the composed prefix and target prompt y^*:

$$p(R_{5:N}^* \mid \mathcal{P}^*, y^*) = \prod_{i=5}^{N} p_\theta(R_i^* \mid R_{<i}^*, y^*), \qquad \theta \text{ frozen.} \tag{13}$$

The generated image is decoded as:

$$I^* = \mathrm{Dec}\left(\sum_{i=1}^{N} \mathrm{Up}(R_i^*)\right). \tag{14}$$

Infinity's bitwise self-correction completes the remaining scales while maintaining consistency with the composed semantic prefix. Although we focus on content–style personalization, the same formulation can support other prompt-definable factors by constructing their corresponding scale-code subspaces.

5 Experiments

5.1 Experimental Setup

Dataset and evaluation protocol. We evaluate Kingfisher on the CSD-100 dataset Nguyen et al. (2025), which contains 100 real images spanning diverse object and appearance concepts. Since our task takes two real images as input, we construct ordered cross-image pairs (I^A, I^B), where I^A is the content image and I^B is the style reference image. For each content image, we sample five style references from different categories while excluding pairs sharing the same object category. This results in 500 ordered content–style pairs. We generate one output for each pair, yielding 500 images per method. For each pair, the target semantic prompt is defined as "a photo of `<content-A>` in `<style-B>` style." We use a fixed seed of 2026 for reproducibility and do not perform best-of-seed selection.

Baselines. We compare Kingfisher with representative personalization and content–style composition methods from both diffusion and visual autoregressive families: DreamBooth-LoRA Ruiz et al. (2023), B-LoRA Frenkel et al. (2024), UnZipLoRA Liu et al. (2025), Inspiration Tree Vinker et al. (2023), and CSD-VAR Nguyen et al. (2025). We use publicly available implementations released by the original authors whenever available, and explicitly document any adaptation required for our cross-image setting or backbone compatibility. The diffusion baselines are implemented on SDXL Podell et al. (2024).

Metrics. We evaluate content preservation, style alignment, and text alignment using five metrics. *CSD-C* and *CSD-S* are computed using the content and style embeddings extracted by the CSD evaluator Nguyen et al. (2025). Specifically, CSD-C is the cosine similarity between the content embedding of the generated image and that of the content image I^A, whereas CSD-S is the cosine similarity between the style embedding of the generated image and that of the style image I^B. *CLIP-I* measures the cosine similarity between the CLIP image features Radford et al. (2021) of the generated image and I^A. *DINO* measures the cosine similarity between DINO visual features Oquab et al. (2023) of the generated image and I^B. Finally, *CLIP-T* measures the cosine similarity between the CLIP image feature of the generated image and the CLIP text feature of the target content–style prompt. Higher values indicate better performance for all metrics.

Table 1: Quantitative comparison on the cross-image CSD-100 benchmark. Best results are shown in **bold** and second-best results are underlined.

	Content Align.		Style Align.		Text Align.
Method	**CSD-C ↑**	**CLIP-I ↑**	**CSD-S ↑**	**DINO ↑**	**CLIP-T ↑**
DreamBooth-LoRA Ruiz et al. (2023)	0.5519	0.7362	0.4207	0.3813	0.2492
B-LoRA Frenkel et al. (2024)	0.4739	0.7182	0.4990	0.4220	0.2683
UnZipLoRA Liu et al. (2025)	**0.5782**	0.7322	0.3673	0.3560	0.2472
Inspiration Tree Vinker et al. (2023)	0.5413	0.7591	0.2897	0.2601	0.2565
CSD-VAR Nguyen et al. (2025)	0.4731	0.7184	0.2812	0.2503	0.2648
Kingfisher (Ours)	0.5519	**0.7900**	**0.5222**	**0.4350**	**0.3105**

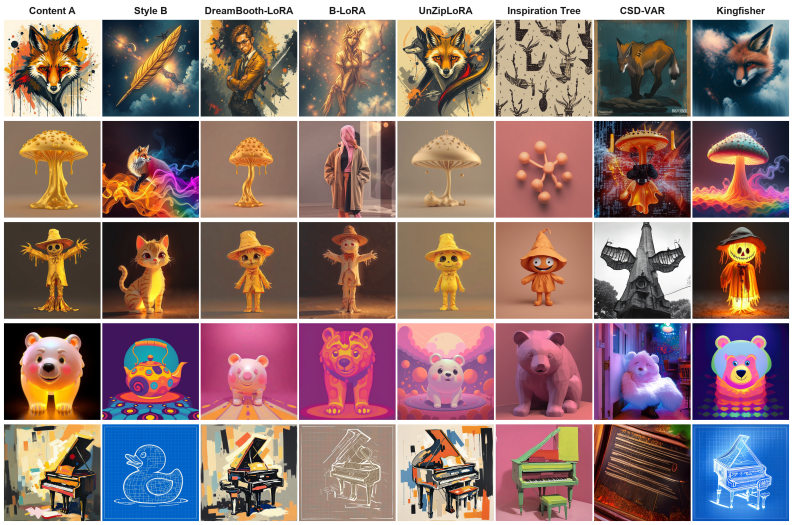

Figure 4: Qualitative comparison on the cross-image CSD-100 benchmark Nguyen et al. (2025). Given a content reference I^A and a style reference I^B, each method aims to preserve the content of I^A while transferring the appearance of I^B. Kingfisher achieves a favorable balance between content preservation and style transfer across diverse content–style combinations.

5.2 Comparison with Prior Methods

Quantitative comparison. Table 1 reports quantitative results over all 500 ordered content–style pairs. Kingfisher achieves the best performance on CLIP-I, CSD-S, DINO, and CLIP-T, showing a strong balance between content preservation and appearance transfer. DreamBooth-LoRA preserves content reasonably well but shows weaker style alignment, which is consistent with the residual entanglement between content and appearance in parameter-efficient personalization. B-LoRA and UnZipLoRA explicitly separate content- and style-related representations and therefore achieve stronger cross-image composition, with B-LoRA favoring style transfer and UnZipLoRA favoring content preservation. Inspiration Tree retains content well but transfers style less effectively in our setting. Among visual autoregressive methods, CSD-VAR explicitly decomposes content and style representations, yet Kingfisher performs substantially better. This suggests that directly manipulating the early residual scale codes provides a more effective way to control the content–style decomposition.

Qualitative comparison. Figure 4 compares representative cross-image compositions under identical content–style pairs. Existing personalization methods typically obtain content and style through learned implicit representations, such as tokens, embeddings, or parameter updates. Since these representations are not explicitly constrained to encode a single factor, content and appearance may remain entangled. As a result, implicit personalization often benefits from multiple reference images of the same object under different backgrounds or conditions to isolate a more consistent object representation, and similarly from multiple examples sharing the same visual style to suppress content-specific cues. With only a single reference image, the learned representation can easily absorb irrelevant appearance or structural information, leading to overfitting to one reference or unnatural compositions when content and style representations are combined. In contrast, Kingfisher explicitly estimates content- and appearance-related components in the early residual scale codes

Table 2: Efficiency and human evaluation. Efficiency is measured on a single A100 GPU. Preparation denotes the pair-specific optimization time for one content–style pair, while inference denotes the time required to generate one image after preparation. Human preference is computed from 600 forced-choice selections.

	Efficiency		Human Evaluation	
Method	**Prep./pair** (s) ↓	**Infer./image** (s) ↓	**Votes** ↑	**Pref.** ↑
DreamBooth-LoRA Ruiz et al. (2023)	1542.21 ± 38.47	11.98 ± 0.91	54	9.0%
B-LoRA Frenkel et al. (2024)	1309.00 ± 27.63	12.88 ± 1.02	168	28.0%
UnZipLoRA Liu et al. (2025)	3463.70 ± 40.28	11.83 ± 0.95	45	7.5%
Inspiration Tree Vinker et al. (2023)	7465.36 ± 216.84	6.56 ± 0.54	39	6.5%
CSD-VAR Nguyen et al. (2025)	171.65 ± 11.82	$\mathbf{1.39 \pm 0.12}$	40	6.7%
Kingfisher (Ours)	$\mathbf{23.54 \pm 1.86}$	1.61 ± 0.15	**254**	**42.3%**

and recomposes them before autoregressive refinement. This leads to more coherent outputs that preserve the object and spatial structure of I^A while transferring the visual appearance of I^B.

Efficiency Analysis Table 2 compares the per-pair computational cost. Unlike methods that optimize model-side weights, LoRA adapters, or learned subject representations, Kingfisher operates directly on the early residual scale codes while keeping the complete Infinity generator frozen. Kingfisher requires only 23.54 seconds of preparation per pair, compared with 171.65 seconds for CSD-VAR. This corresponds to more than a $10\times$ reduction in preparation cost. Kingfisher also requires only 1.61 seconds to generate an output, remaining close to the fastest autoregressive baseline while being substantially faster than the diffusion-based methods.

Human Evaluation We conducted a blind forced-choice study with 20 participants on 600 responses to compare Kingfisher with existing five methods. For each pair of content-style, participants selected the output that best preserved the content image, transferred the style reference, and maintained overall image quality. As shown in Table 2, Kingfisher receives the highest human preference among the evaluated methods, followed by B-LoRA. The results indicate that participants more frequently prefer Kingfisher when jointly considering content preservation, appearance transfer, and overall visual quality, consistent with the trends observed in our qualitative and automatic evaluations.

6 Discussion

Our results highlight a different perspective on personalization in visual autoregressive models. Many existing personalization approaches represent content and style implicitly through learned tokens, embeddings, adapters, or parameter updates, where the two factors may remain entangled within the learned representation. In contrast, Kingfisher explicitly models each early residual scale in terms of content- and appearance-related components and directly constrains their roles through prompt-conditioned scale subspaces. This formulation provides an explicit mechanism for analyzing and recomposing content and appearance within the native residual scale trajectory of a frozen generator. Our scale-wise analysis further shows that these factors are not naturally separated across individual scales; instead, the early scales form an entangled semantic bottleneck that jointly determines object structure and global appearance. By operating only within this compact early prefix, Kingfisher also substantially reduces the personalization cost compared with optimization-heavy alternatives.

7 Conclusion

We introduced Kingfisher, a fast personalization framework for visual autoregressive models that operates directly on the residual scale trajectory of a frozen pretrained generator. By explicitly decomposing and recomposing content- and appearance-related components within the early semantic prefix, Kingfisher achieves effective content–style personalization without model fine-tuning. Experiments on CSD-100 dataset show that Kingfisher achieves strong content–style composition and substantially reduces personalization cost. These results demonstrate that native scale trajectories can provide an efficient interface for personalized visual autoregressive generation without model fine-tuning.

AI USE STATEMENT

ChatGPT was used solely for grammar correction and language polishing. All technical methodology, experiments, and results were developed, conducted, and verified by the authors.

ETHICS STATEMENT

Our human evaluation involves voluntary and anonymous preference judgments on generated images. No personal identifiers or sensitive information were collected, and the responses were used solely to evaluate perceptual preferences among the compared methods. We follow the ICLR Code of Ethics throughout the study.

REPRODUCIBILITY STATEMENT

We provide detailed descriptions of the CSD-100 evaluation protocol, baseline configurations, evaluation metrics, implementation settings, and hyperparameters in Sec. 5. Additional qualitative results, ablation studies, and the exact neutral and object-conditioned prompt templates used to construct the prompt-conditioned subspaces are provided in the appendix. These details are intended to facilitate reproduction of the reported results.

REFERENCES

Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4432–4441, 2019.

Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *International conference on machine learning*, pp. 1691–1703. PMLR, 2020.

P Kingma Diederik and Welling Max. An introduction to variational autoencoders. *Foundations and Trends® in Machine Learning*, 12(4):307–392, 2019.

Sara Dorfman, Dana Cohen-Bar, Rinon Gal, and Daniel Cohen-Or. 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.

Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12873–12883, 2021.

Yarden Frenkel, Yael Vinker, Ariel Shamir, and Daniel Cohen-Or. Implicit style-content separation using b-lora. In *European Conference on Computer Vision*, pp. 181–198. Springer, 2024.

Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022.

Yuchao Gu, Xintao Wang, Jay Zhangjie Wu, Yujun Shi, Yunpeng Chen, Zihan Fan, Wuyou Xiao, Rui Zhao, Shuning Chang, Weijia Wu, et al. Mix-of-show: Decentralized low-rank adaptation for multi-concept customization of diffusion models. *Advances in Neural Information Processing Systems*, 36:15890–15902, 2023.

Jian Han, Jinlai Liu, Yi Jiang, Bin Yan, Yuqi Zhang, Zehuan Yuan, Bingyue Peng, and Xiaobing Liu. Infinity: Scaling bitwise autoregressive modeling for high-resolution image synthesis. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 15733–15744, 2025.

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

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *International Conference on Learning Representations*, 1(2):3, 2022.

Chen Jin, Ryutaro Tanno, Amrutha Saseendran, Tom Diethe, and Philip Alexander Teare. An image is worth multiple words: Discovering object level concepts using multi-concept prompt learning. In *Forty-first International Conference on Machine Learning*, 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, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. *Advances in neural information processing systems*, 33:12104–12114, 2020.

Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1931–1941, 2023.

Kyoungmin Lee, Jihun Park, Jongmin Gim, Wonhyeok Choi, Kyumin Hwang, Jaeyeul Kim, and Sunghoon Im. A training-free style-personalization via svd-based feature decomposition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 506–516, 2026.

Zhen Li, Mingdeng Cao, Xintao Wang, Zhongang Qi, Ming-Ming Cheng, and Ying Shan. Photomaker: Customizing realistic human photos via stacked id embedding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8640–8650, 2024.

Chang Liu, Viraj Shah, Aiyu Cui, and Svetlana Lazebnik. Unziplora: Separating content and style from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16776–16785, 2025.

Jian Ma, Junhao Liang, Chen Chen, and Haonan Lu. Subject-diffusion: Open domain personalized text-to-image generation without test-time fine-tuning. In *ACM SIGGRAPH 2024 Conference Papers*, pp. 1–12, 2024.

Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6038–6047, 2023.

Quang-Binh Nguyen, Minh Luu, Quang Nguyen, Anh Tran, and Khoi Nguyen. Csd-var: Content-style decomposition in visual autoregressive models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 17013–17023, 2025.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

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. In *International Conference on Learning Representations*, volume 2024, pp. 1862–1874, 2024.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763, 18–24 Jul 2021.

Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019.

Scott Reed, Aäron Oord, Nal Kalchbrenner, Sergio Gómez Colmenarejo, Ziyu Wang, Yutian Chen, Dan Belov, and Nando Freitas. Parallel multiscale autoregressive density estimation. In *International conference on machine learning*, pp. 2912–2921. PMLR, 2017.

Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a stylegan encoder for image-to-image translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2287–2296, 2021.

Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 22500–22510, 2023.

Viraj Shah, Nataniel Ruiz, Forrester Cole, Erika Lu, Svetlana Lazebnik, Yuanzhen Li, and Varun Jampani. Ziplora: Any subject in any style by effectively merging loras. In *European Conference on Computer Vision*, pp. 422–438. Springer, 2024.

Jing Shi, Wei Xiong, Zhe Lin, and Hyun Joon Jung. Instantbooth: Personalized text-to-image generation without test-time finetuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8543–8552, 2024.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

Keyu Tian, Yi Jiang, Zehuan Yuan, Bingyue Peng, and Liwei Wang. Visual autoregressive modeling: Scalable image generation via next-scale prediction. *Advances in neural information processing systems*, 37:84839–84865, 2024.

Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or. Designing an encoder for stylegan image manipulation. *ACM Transactions on Graphics (TOG)*, 40(4):1–14, 2021.

Aaron Van den Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves, et al. Conditional image generation with pixelcnn decoders. *Advances in neural information processing systems*, 29, 2016.

Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Yael Vinker, Andrey Voynov, Daniel Cohen-Or, and Ariel Shamir. Concept decomposition for visual exploration and inspiration. *ACM Transactions on Graphics (TOG)*, 42(6):1–13, 2023.

Andrey Voynov, Qinghao Chu, Daniel Cohen-Or, and Kfir Aberman. p+: Extended textual conditioning in text-to-image generation. *arXiv preprint arXiv:2303.09522*, 2023.

Qixun Wang, Xu Bai, Haofan Wang, Zekui Qin, Anthony Chen, Huaxia Li, Xu Tang, and Yao Hu. Instantid: Zero-shot identity-preserving generation in seconds. *arXiv preprint arXiv:2401.07519*, 2024.

Xierui Wang, Siming Fu, Qihan Huang, Wanggui He, and Hao Jiang. Ms-diffusion: Multi-subject zero-shot image personalization with layout guidance. In *International Conference on Learning Representations*, volume 2025, pp. 95118–95146, 2025.

Yuxiang Wei, Yabo Zhang, Zhilong Ji, Jinfeng Bai, Lei Zhang, and Wangmeng Zuo. Elite: Encoding visual concepts into textual embeddings for customized text-to-image generation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 15943–15953, 2023.

Yuxiang Wei, Yiheng Zheng, Yabo Zhang, Ming Liu, Zhilong Ji, Lei Zhang, and Wangmeng Zuo. Personalized image generation with deep generative models: A decade survey. *Computational Visual Media*, 2025.

Zongze Wu, Dani Lischinski, and Eli Shechtman. Stylespace analysis: Disentangled controls for stylegan image generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12863–12872, 2021.

Yiyan Xu, Jinghao Zhang, Alireza Salemi, Xinting Hu, Wenjie Wang, Fuli Feng, Hamed Zamani, Xiangnan He, and Tat-Seng Chua. Personalized generation in large model era: A survey. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 24607–24649, 2025.

Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*, 2023.

Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2(3):5, 2022.

APPENDIX

A ADDITIONAL SCALE-WISE ANALYSIS

Table 3 reports the quantitative results of the controlled prefix-reconstruction and prompt-routing experiments described in Sec. 3.

Table 3: **Quantitative scale-wise analysis.** Results are averaged over 100 real images and 200 prompt-routing pairs. For prefix reconstruction, CLIP-I and DINO-I measure similarity to the input image, while LPIPS measures perceptual reconstruction distance. For prompt routing, CLIP-T measures alignment with Prompt_2, while DINO-P1 measures similarity to the all-Prompt_1 reference.

Experiment	Range	CLIP-I ↑	DINO-I ↑	LPIPS ↓	CLIP-T ↑	DINO-P1 ↑
Prefix recon.	R_1	0.42	0.36	0.58	–	–
Prefix recon.	$R_{1:2}$	0.61	0.54	0.41	–	–
Prefix recon.	$R_{1:3}$	0.78	0.72	0.26	–	–
Prefix recon.	$R_{1:4}$	0.88	0.84	0.16	–	–
Prefix recon.	$R_{1:7}$	0.93	0.89	0.11	–	–
Prompt routing	$R_{10:13}$	–	–	–	0.28	0.74
Prompt routing	$R_{5:13}$	–	–	–	0.34	0.70
Prompt routing	$R_{1:3}$	–	–	–	0.76	0.39
Prompt routing	$R_{2:4}$	–	–	–	0.81	0.35
Prompt routing	$R_{1:5}$	–	–	–	0.84	0.31

B CONDITION-PROMPT CONSTRUCTION

Kingfisher constructs the prompt-conditioned scale subspaces using two prompt sets. The neutral prompts describe realistic images with ordinary colors, illumination, texture, and background, providing a neutral appearance reference. The object-conditioned prompts describe the target object through multiple semantic formulations, providing diverse object-related directions while reducing sensitivity to a particular wording. Table 4 lists the prompt templates used in our experiments, where `<object>` is replaced by the object category of each input image.

Table 4: Neutral and object-conditioned prompt templates used to construct the prompt-conditioned scale subspaces.

Type	Prompt template
Neutral	"a plain realistic photograph with natural colors and ordinary daylight" "a standard documentary photograph with neutral balanced lighting" "a clean product photograph with a simple background and natural texture" "a realistic snapshot with balanced colors and soft natural illumination" "a straightforward studio photograph with ordinary materials and neutral tones"
Object	"a clearly recognizable `<object>`" "a detailed `<object>`" "a centered `<object>`" "a close-up view of a `<object>`" "a `<object>` with a clear silhouette" "a realistic photograph of a `<object>`"

The neutral predictions are used to construct the neutral-style affine subspace $\mathcal{N}_i$, while the object-conditioned predictions are centered relative to the neutral reference and used to construct the object-content subspace $\mathcal{C}_i$. These prompts are used only to query the frozen generator and introduce no learned textual embeddings or additional trainable parameters.

C DETAILS OF HUMAN EVALUATION

We conduct a human preference study to complement the automatic evaluation by presenting the generated results from all evaluated methods side by side. Each question presents one content–style pair consisting of the content reference I^A, the style reference I^B, and six anonymous outputs

generated by Kingfisher, DreamBooth-LoRA Ruiz et al. (2023), B-LoRA Frenkel et al. (2024), UnZipLoRA Liu et al. (2025), Inspiration Tree Vinker et al. (2023), and CSD-VAR Nguyen et al. (2025). Participants are asked to select the single output that best jointly preserves the object identity and spatial structure of I^A, adopts the visual appearance of I^B, and maintains overall image quality. Method identities are hidden, and both the order of the content–style pairs and the order of the seven method outputs within each question are independently randomized for each questionnaire version. To reduce fatigue, the study is divided into 10 questionnaire versions, each containing 30 unique pairs, covering 300 of the 500 benchmark pairs in total. We collected responses from 20 participants, resulting in 600 total selections. Depending on the assigned questionnaire version, each evaluated content–style pair receives between one and three independent evaluations. We aggregate all valid selections and report the percentage of preferences received by each method.

D Ablation Studies

Early-scale prefix composition. We investigate how progressively rectifying the early residual scales affects the content–style trade-off. Table 5 shows that increasing the number of rectified scales consistently improves style alignment while reducing content preservation. The original prefix $[A_1, A_2, A_3, A_4]$ achieves the strongest content scores but the weakest style scores. Introducing R_1^* and R_4^* improves style alignment, and further rectifying R_2 provides a more balanced trade-off between content and style while achieving the highest CLIP-T score. Rectifying R_3 as well yields the strongest style alignment but at the cost of substantially lower content similarity. We therefore adopt $[R_1^*, R_2^*, A_3, R_4^*]$ as the default configuration to maintain a balanced trade-off between content preservation and appearance transfer.

Table 5: Ablation of early-scale prefix composition. Best results are shown in bold and second-best results are underlined.

Prefix configuration	CSD-C ↑	CLIP-I ↑	CSD-S ↑	DINO ↑	CLIP-T ↑
$[A_1, A_2, A_3, A_4]$	**0.7774**	**0.9370**	0.2922	0.2720	0.3017
$[R_1^*, A_2, A_3, R_4^*]$	<u>0.6281</u>	<u>0.8392</u>	0.4342	0.3751	<u>0.3119</u>
$[\mathbf{R_1^*}, \mathbf{R_2^*}, \mathbf{A_3}, \mathbf{R_4^*}]$	0.5504	0.7922	<u>0.5210</u>	<u>0.4367</u>	**0.3127**
$[R_1^*, R_2^*, R_3^*, R_4^*]$	0.4979	0.7416	**0.6156**	**0.5416**	0.2959

Structural correction strength. We next study the coefficient β, which controls the contribution of the content correction at the fourth residual scale, $R_4^* = B_4 + \beta(C_4^A - C_4^B)$. Table 6 reveals a clear trade-off: as β increases, CSD-C and CLIP-I consistently improve, whereas CSD-S and DINO decrease. Larger values therefore favor content preservation, while smaller values retain stronger appearance alignment. We use $\beta = 0.5$ as an intermediate operating point that balances these two tendencies without strongly favoring either content or style.

Table 6: Sensitivity to the structural correction coefficient β. Best results are shown in bold and second-best results are underlined.

β	CSD-C ↑	CLIP-I ↑	CSD-S ↑	DINO ↑	CLIP-T ↑
0.00	0.5436	0.7784	**0.5353**	**0.4597**	0.3068
0.25	0.5471	0.7866	<u>0.5333</u>	<u>0.4521</u>	0.3104
0.50	0.5504	0.7922	0.5210	0.4367	0.3127
0.75	<u>0.5551</u>	<u>0.7968</u>	0.5161	0.4239	<u>0.3130</u>
1.00	**0.5613**	**0.8027**	0.5063	0.4096	**0.3144**

Decomposition constraints. Table 7 evaluates the contribution of each term in the decomposition objective. Removing $\mathcal{L}_{\text{sty}}$ mainly degrades style alignment, confirming its role in isolating the appearance component. Removing the content alignment or removal constraints primarily reduces content preservation, as the estimated content component is no longer sufficiently constrained to represent and isolate object-related information. Without $\mathcal{L}_{\text{rec}}$, both content and style performance decrease substantially because information is no longer preserved during decomposition. Finally, removing $\mathcal{L}_{\text{orth}}$ causes the content and style components to overlap, leading to cross-factor leakage

and moderate degradation in both aspects. Overall, the full objective provides the most consistent balance across content preservation, style transfer, and text alignment.

Table 7: Ablation of the decomposition objective. Best results are shown in bold and second-best results are underlined.

Configuration	CSD-C ↑	CLIP-I ↑	CSD-S ↑	DINO ↑	CLIP-T ↑
w/o $\mathcal{L}_{\text{sty}}$	<u>0.5478</u>	<u>0.7895</u>	0.4726	0.3914	0.3048
w/o $\mathcal{L}_{\text{align}}$	0.5127	0.7568	<u>0.5073</u>	<u>0.4218</u>	0.3035
w/o $\mathcal{L}_{\text{remove}}$	0.5019	0.7446	0.4981	0.4127	0.3009
w/o $\mathcal{L}_{\text{rec}}$	0.4742	0.7165	0.4386	0.3597	0.2863
w/o $\mathcal{L}_{\text{orth}}$	0.5264	0.7698	0.4935	0.4062	<u>0.3074</u>
Full objective	**0.5504**	**0.7922**	**0.5210**	**0.4367**	**0.3127**

Prefix composition variants visualization. Figure 5 provides additional qualitative comparisons of the early-scale prefix configurations under two structural correction strengths, $\beta = 0.5$ and $\beta = 1.0$. Given the same content image I^A and style image I^B, we compare $[R_1^*, A_2, A_3, R_4^*]$, $[R_1^*, R_2^*, A_3, R_4^*]$, and $[R_1^*, R_2^*, R_3^*, R_4^*]$. These examples further illustrate the content–appearance trade-off observed in the quantitative ablation: progressively rectifying more early scales strengthens appearance transfer, while increasing β places greater emphasis on the structural information from the content reference.



Figure 5: **Additional qualitative ablation of prefix composition.** From left to right: content reference I^A, style reference I^B, $[R_1^*, A_2, A_3, R_4^*]$ with $\beta = 0.5$, $[R_1^*, R_2^*, A_3, R_4^*]$ with $\beta = 0.5$, $[R_1^*, R_2^*, R_3^*, R_4^*]$ with $\beta = 0.5$, $[R_1^*, A_2, A_3, R_4^*]$ with $\beta = 1.0$, $[R_1^*, R_2^*, A_3, R_4^*]$ with $\beta = 1.0$, and $[R_1^*, R_2^*, R_3^*, R_4^*]$ with $\beta = 1.0$.

Style-removal visualization. Figure 6 visualizes the effect of the estimated appearance component by comparing each input image with its style-removed representation $R - S$. Removing S suppresses appearance-specific cues and moves the representation toward the neutral-appearance space, while retaining the dominant object structure. These examples provide a qualitative diagnostic of the appearance component estimated by our prompt-conditioned decomposition.

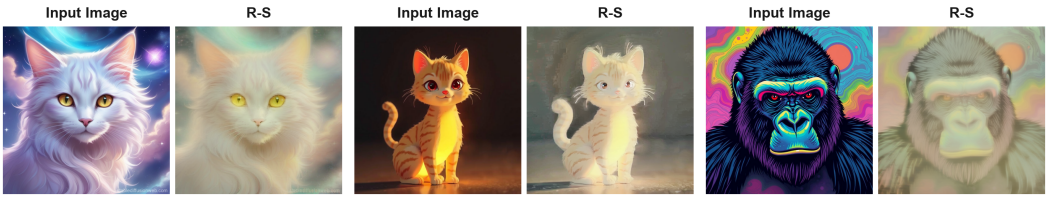

Figure 6: **Visualization of style removal.** Each pair shows an input image and the corresponding style-removed result $R - S$. The operation suppresses appearance-specific information while retaining the underlying object content and structure.

E Additional Qualitative Evaluation

Figure 7 presents additional qualitative comparisons with prior methods across diverse content–style pairs. The results further show that Kingfisher achieves a favorable balance between preserving the content structure of I^A and transferring the visual appearance of I^B.

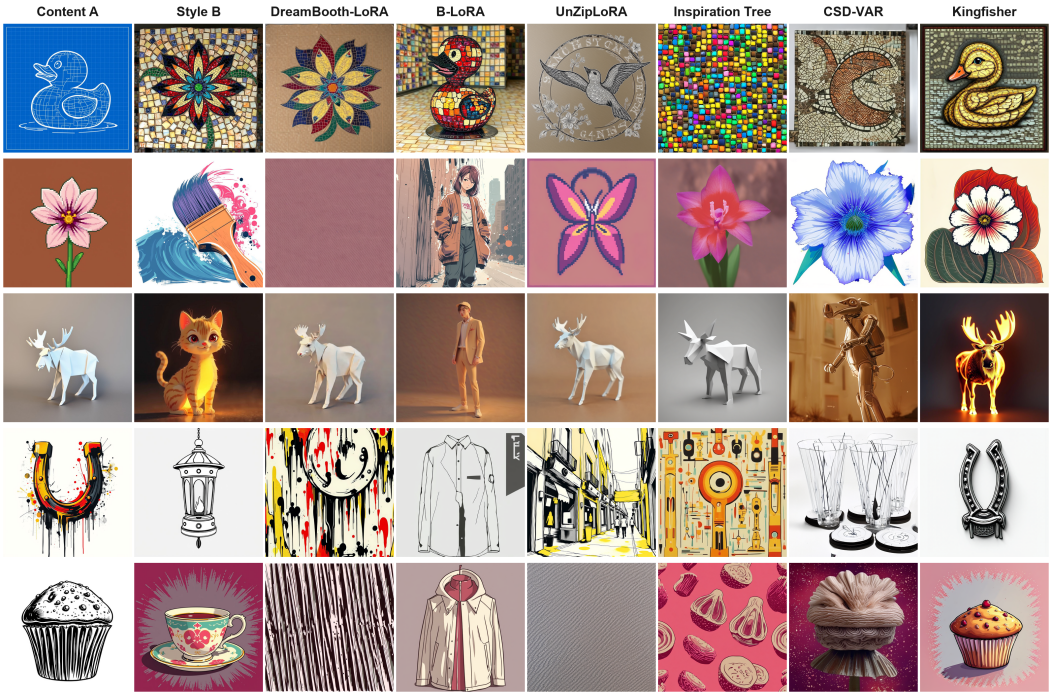

Figure 7: **Additional qualitative comparison with prior methods.** Given a content reference I^A and a style reference I^B, we compare Kingfisher with the evaluated baselines across additional content–style combinations.

F Qualitative Results

Figure 8 presents qualitative results on the CSD-100 benchmark. Given a content reference I^A and a style reference I^B, Kingfisher consistently preserves the main object structure of the content image while transferring the visual appearance of the style reference across diverse content–style combinations.

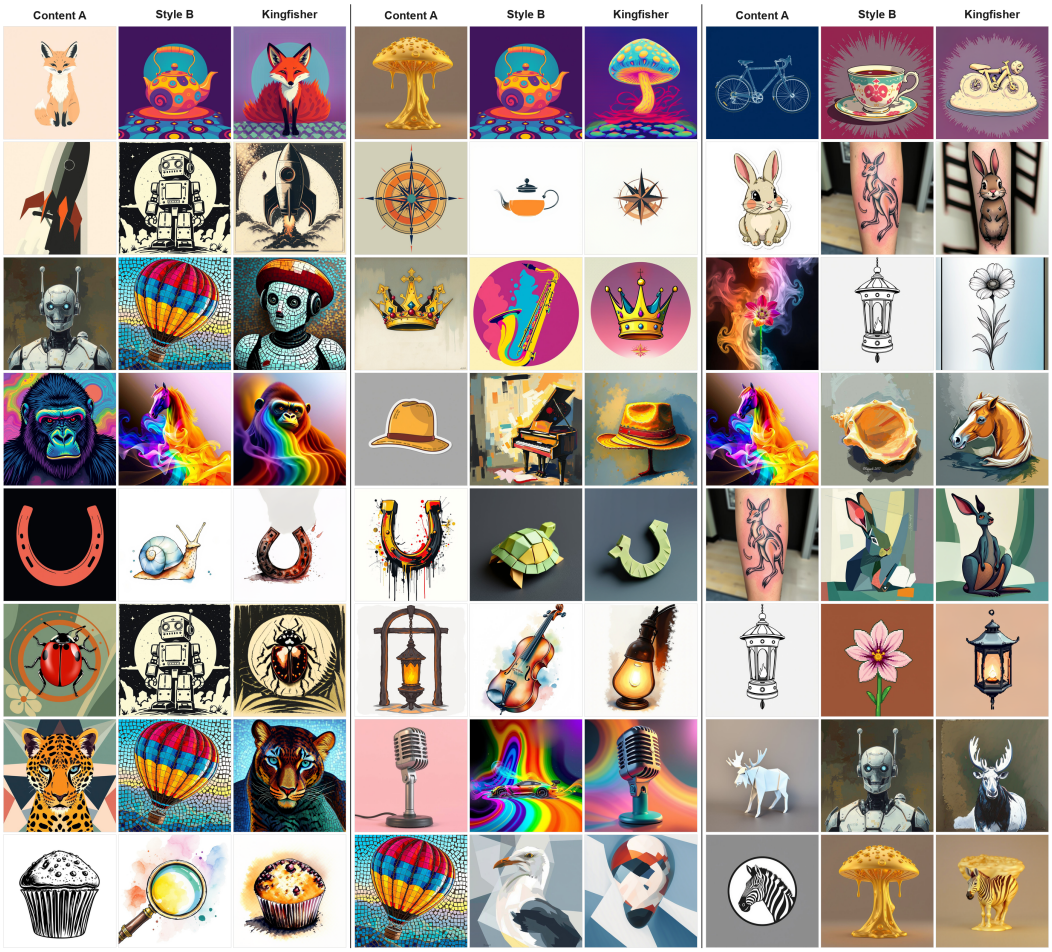

Figure 8: **Qualitative results of Kingfisher on CSD-100.** Each example shows the content reference I^A, the style reference I^B, and the corresponding Kingfisher output.

G Implementation Details

For **Kingfisher**, Kingfisher is built on the publicly available pretrained **Infinity-2B** model Han et al. (2025), which is kept entirely frozen throughout all experiments. Infinity represents an image using $N = 13$ residual scales. Kingfisher performs prompt-conditioned decomposition and composition only on the early semantic prefix R_1–R_4, while the remaining scales $R_5, \ldots, R_{13}$ are autoregressively predicted by the pretrained Infinity-2B generator. We use five neutral prompts and six object prompts and retain the first four singular directions for both prompt-conditioned subspaces. The content and appearance components are optimized for 250 steps using Adam with a learning rate of 3×10^{-2}. We set

$$\lambda_{\text{sty}} = 1.0, \qquad \lambda_{\text{align}} = 0.5, \qquad \lambda_{\text{remove}} = 0.75, \qquad \lambda_{\text{rec}} = 0.05, \qquad \lambda_{\text{orth}} = 0.25.$$

Our default composed prefix is

$$[R_1^*, R_2^*, A_3, R_4^*], \tag{15}$$

where

$$R_1^* = A_1 - S_1^A + S_1^B, \qquad R_2^* = A_2 - S_2^A + S_2^B,$$

and

$$R_4^* = B_4 + \beta(C_4^A - C_4^B), \qquad \beta = 0.5.$$

Classifier-free guidance is set to 3.0. No model weights, adapters, K–V representations, or subject-specific embeddings are updated during personalization.

For **B-LoRA**, we optimize one rank-64 representation for each reference image for 1,000 steps with a learning rate of 5×10^{-5}. Each image is associated with a generic learned identifier "`A [vN]`". For a pair (I^A, I^B), we combine the official content block from I^A with the style block from I^B, using "`A [vA] in [vB] style`" for generation.

For **DreamBooth-LoRA**, we learn role-specific content and style LoRA representations from each reference. The content representation is trained using "`a photo of [cN]`", while the appearance representation uses "`an image in [sN] style`". At inference, the content LoRA from I^A and the style LoRA from I^B are activated jointly, and the target is generated using "`a photo of [cA] in [sB] style`".

For **UnZipLoRA**, we use its SDXL implementation with rank 64 and 600 optimization steps per reference. Cross-image inference uses the content LoRA and merger learned from I^A, together with the style LoRA and merger learned from I^B.

The original **Inspiration Tree** implementation targets SD1.x; we therefore use an SDXL port following its original optimization principle. For each reference, four candidate trees are optimized for 200 steps, selected using image-based coherence and distinctness, and the selected initialization is replayed to 1,000 steps. During cross-image composition, the learned children from I^A and I^B are used as the content and style references, respectively.

For **CSD-VAR**, we report the official Switti backbone implementation. CSD-VAR decomposes each source image independently as

$$I^A \to (t_C^A, t_S^A), \qquad I^B \to (t_C^B, t_S^B), \tag{16}$$

and generates the cross-image result using the content representation t_C^A and style representation t_S^B. We retain the semantic caption and initializer conditioning used by the released CSD-VAR pipeline.

H LIMITATIONS

Kingfisher is less effective when the reference appearance is visually weak or insufficiently distinctive. In particular, styles with limited color variation, texture, illumination, or other characteristic appearance cues may induce only a small deviation from the neutral appearance space, making the estimated style component less discriminative and resulting in weaker perceptual transfer (as illustrated in Fig. 9). In addition, our decomposition relies on prompt-conditioned object directions; therefore, its effectiveness depends on how accurately the prompts describe the target content and how well the frozen generator responds to these descriptions.

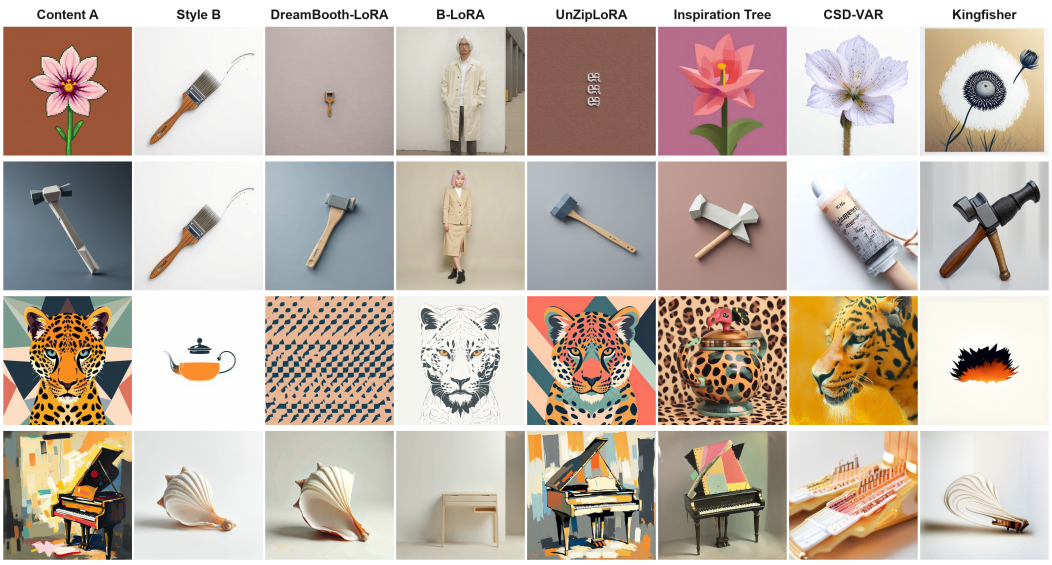

Figure 9: **Challenging cases with abstract or ambiguous style references.** When the style reference contains weak, highly abstract, or insufficiently distinctive visual cues, its appearance is difficult to isolate from a single image. Consequently, both existing baselines and Kingfisher may exhibit weaker or less consistent style transfer.