APPEARANCE POINTERS: MULTIMODAL REGION CONTROL OF DIFFUSION TRANSFORMERS

Anonymous authors
Paper under double-blind review

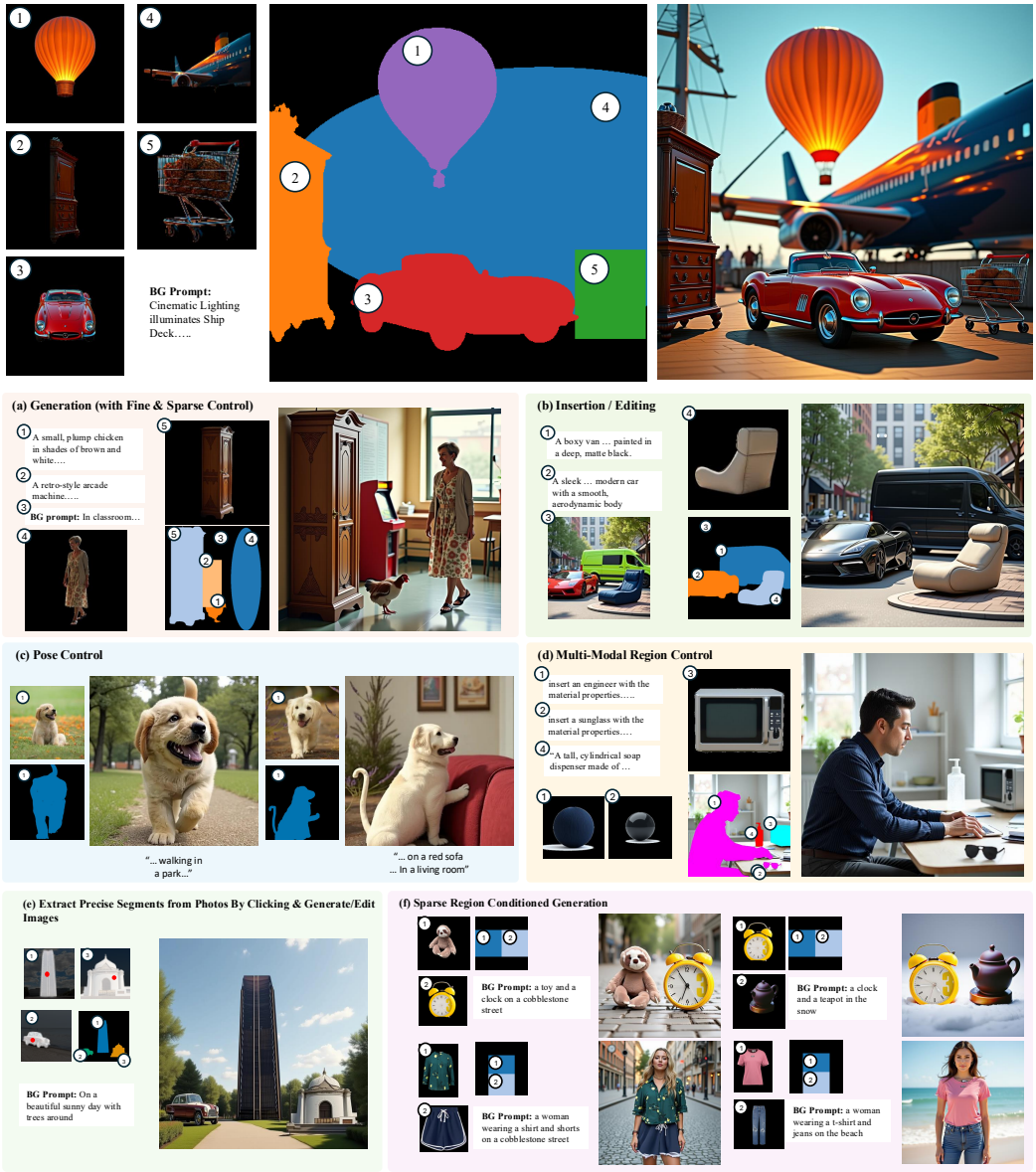

Figure 1: **Appearance Pointers enable precise, multimodal, region-aware image generation in a single denoising pass**. Given spatial masks and heterogeneous conditioning signals — reference images or text descriptions — our region correspondence network produces **compact pointer tokens** that **route a Diffusion Transformer** toward the correct appearance cues at the appropriate spatial locations. Our work supports: **(a)** generation from fine or sparse regional layouts; **(b)** object insertion with material and style fidelity; **(c)** pose-conditioned generation; **(d)** simultaneous multi-modal region control, combining image- and text-based descriptions in a single scene; **(e)** composing elements from different images via simple clicks (shown in red) that segment the desired objects; and **(f)** sparse region control when fine regions are not provided.

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.

ABSTRACT

Creative professionals rarely want a single global description of an image; they want region-level control over the material, identity, and placement of individual objects, which text prompting alone cannot reliably deliver. Diffusion Transformers (DiTs) natively ingest heterogeneous text and image tokens, yet provide no mechanism to decide *where* and *how* each token should shape the output. We introduce *appearance pointers*, compact tokens that route a frozen DiT to the correct appearance cue at the correct region by aligning text or reference-image inputs with user-specified masks. A region correspondence network produces these pointers, and a depth-wise aggregation step fuses many regions into a single canvas, so multiple heterogeneous regions are generated in one denoising pass without inflating the token count. Unlike prior region-control methods — tied to a single modality or one region at a time — *AppearancePointers* is the first to condition the *same* region on image *and* text simultaneously, with one set of weights and only $\sim 7.5\%$ added parameters. Across text- and image-conditioned benchmarks, and zero-shot on real-world data, our single model matches or surpasses modality-specific state-of-the-art methods, offering a simple, extensible interface for precise multimodal region control.

1 INTRODUCTION

Recent advances in generative modeling have dramatically expanded the creative possibilities of machine-generated imagery. Artists, designers, and filmmakers can now synthesize realistic and diverse scenes by simply providing a text description. Yet, despite their expressive potential, current systems remain difficult to control in practice. Creative professionals often operate with specific visual intentions — precise materials, object layouts, and stylistic details that define the desired outcome. In contrast, text prompting provides only indirect and often unpredictable control, requiring extensive trial and error to achieve the intended composition. This gap between human intention and model controllability limits the integration of generative models into real-world creative pipelines.

Diffusion Transformers (DiTs) — now core to many state-of-the-art image generators — offer an appealing pathway toward richer controllability because they can natively ingest heterogeneous token streams, including text embeddings and image tokens. However, they lack a mechanism to determine where and how these modality-specific tokens should influence the generated image. Simply providing more text or image tokens does not convey the spatial intent of the user, nor does it specify which appearance cues should be used in which regions. As a result, structured and localized user intent, expressed through text or reference images, remains difficult to translate into region-aware control.

Existing controllable generation approaches attempt to address this through attention manipulation Zhou et al. (2025a;b); Zhang et al. (2025b), specialized adapters Zhang et al. (2023); Mou et al. (2023); Peng et al. (2024); Zhao et al. (2024); Mo et al. (2024); Jia et al. (2024); Lukovnikov & Fischer (2025), noise initialization strategies Mao et al. (2023b;a); Sun et al. (2024), or gradient-based guidance during inference Xie et al. (2023); Chen et al. (2024a); Sun et al. (2024); Zhao et al. (2023); Xiao et al. (2023); Couairon et al. (2023); Phung et al. (2023); Lukovnikov & Fischer (2025). The majority of existing work is developed for older U-Net-based diffusion models and adopts inference-time approaches that are often slow and non-robust.

Moreover, most are constrained to a single conditioning modality (e.g., text-only or image-only) and do not seamlessly combine heterogeneous cues Saha et al. (2025); Zhang et al. (2025b). Even with the adoption of recent DiT backbones, these limitations restrict the practical use of diffusion models in tasks requiring multimodal and spatially grounded control.



The figure to the right visualizes this effect: reference images are passed to a DiT to generate the image in the left column, and the yellow/green overlays mark the reference

tokens attended by probe tokens (yellow/green stars) placed in the generated image. The bottom row — a vanilla baseline that simply passes reference-image tokens to the DiT — fails to route appearance to the intended regions, whereas our method (top row) attends to the correct region. We call these *AppearancePointers* — compact tokens produced by a small region correspondence network that fuses each text or image prompt with its spatial mask. Unlike adapters that inject stored appearance features or attention-manipulation methods that reweight existing maps, appearance pointers instead route the DiT's own image and text tokens by telling it *where* each should apply — preserving the backbone's flexibility while adding a lightweight, modular interface for localized control. A spatial aggregation mechanism then fuses many regions into one representation, letting the model satisfy several image- or text-described regions *within a single denoising pass*. Crucially, this allows a single region to be conditioned on image *and* text at once (fig. 1d, Table 1).

To train and evaluate this model, we created a synthetic dataset with image and text description for multiple elements within the same image. When regions are described only through text, our approach is the best or second best in five of the six metrics. When images are used for regional description, our approach surpasses MSDiffusion Wang et al. (2025b) & DreamRenderer Zhou et al. (2025a) in region adherence and identity preservation. Beyond these quantitative gains, appearance pointers offer a simple and extensible path toward precise, multimodal, region-aware control in generative modeling, enabling users to specify not only what should appear in an image, but where and how they should be realized. To sum up, our main contributions are:

- We introduce *AppearancePointers*, a compact representation that routes diffusion transformers to their appropriate regional signals in the form of images, text, and masks.
- Our modular and flexible approach extends to multiple edit capabilities including: (a) region controlled generation, (b) insertion, (c) pose control, and (d) multi-modal region-controlled generation for multiple regions in a single denoising process.
- Our design computes appearance pointers *once per generation* rather than per denoising step and adds only $\sim 7.5\%$ parameters to the base model, keeping the overhead negligible.
- We also propose a dataset *AppearancePointers*-37K containing region text descriptions, appearance images from novel views, and an automatic generation scheme to generate more data.

2 Related Works

We cover the closest region-based control methods here and defer broader works to Appendix A. Table 1 positions *AppearancePointers* against the state of the art: unlike modality- or task-specific prior work, we target diverse controllable workflows within a *single* framework.

Method	Image	Text	Region Control		Insertion	Generation	MultiModal
			Fine	Sparse			
InsertAnything Song et al. (2025)	✓	×	✓	✓	✓	×	×
MS-Diffusion Wang et al. (2025b)	✓	×	×	✓	×	✓	×
Sigma-Gen Saha et al. (2025)	✓	×	✓	✓	×	✓	×
DreamRenderer Zhou et al. (2025a)	✓	✓	×	✓	×	✓	×
InstanceDiffusion Wang et al. (2024)	×	✓	✓	✓	×	✓	×
Seg2Any Zhang et al. (2025b)	×	✓	✓	×	×	✓	×
Ours	✓	✓	✓	✓	✓	✓	✓

Table 1: **Region-conditioned generation methods across key capabilities.** *Image*/*Text*: region condition described by image and/or text; *Fine*/*Sparse*: precise-shape vs. coarse (box) regions; *Insertion*: placing an object into an existing scene; *Generation*: full synthesis from regional descriptions; *MultiModal*: image *and* text for the *same* region in a single pass (e.g. the material assignment in Fig. 2). Unlike all prior works, *AppearancePointers* supports every capability simultaneously.

Bounding boxes and layouts. Region control from boxes or segmentation maps has been addressed by *training-free* methods via noise initialization, cross- and self-attention manipulation, or gradient-based attention alignment Mao & Wang (2023); Xie et al. (2023); Shirakawa & Uchida (2024); Chen et al. (2024a); Taghipour et al. (2025); Balaji et al. (2022); Liu et al. (2024); Zhao et al. (2023); Xiao et al. (2023); Sun et al. (2024); Mao et al. (2023b;a); Zhou et al. (2025b), and by *training-based* modules that fine-tune the backbone (e.g., GLIGEN Li et al. (2023), SSMG Jia et al. (2024), LayoutDiffusion Zheng et al. (2023)), with recent extensions to auto-regressive generation Cui et al. (2025); Zheng et al. (2025). These typically rely on coarse boxes and lack precise shape control. A parallel line strengthens or aligns attention within masked regions, either training-free He et al. (2023); Kim et al. (2023); Couairon et al. (2023); Phung et al.

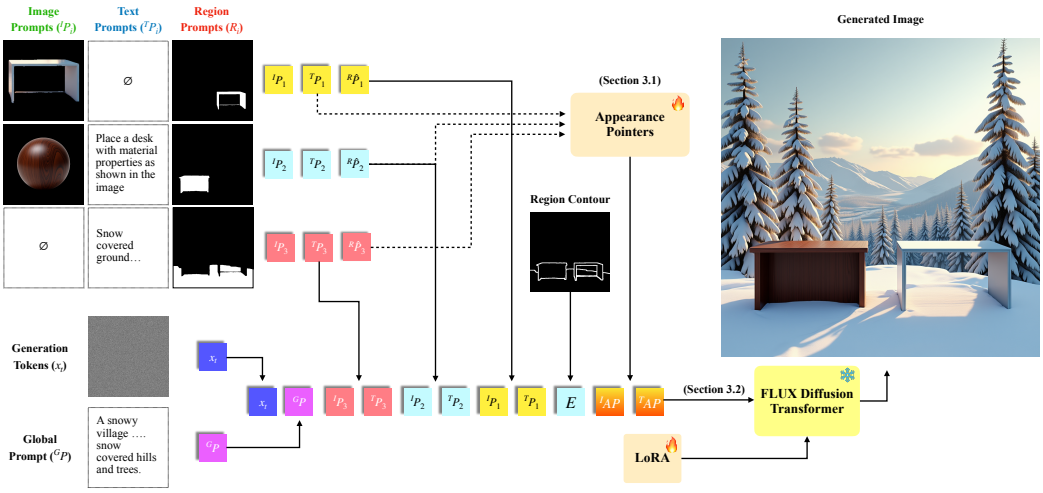

Figure 2: **Overview of AppearancePointers**. Our framework converts heterogeneous inputs (text, images, and spatial masks) into *appearance pointers* (yellow). After encoding via FLUX VAE and T5 text encoder, multimodal signals from image ($^{I}P_i$) and text ($^{T}P_i$), are fused with their spatial masks (R_i) to form Appearance Pointers ($^{I}\mathcal{AP}$, $^{T}\mathcal{AP}$), dictating exactly *which* features map to specific regions. During sampling, the diffusion transformer receives noisy tokens, text/image tokens, and the appearance pointers as joint inputs to then synthesize a coherent and spatially aligned output that respects all region-specific instructions across modalities. See section 3 for more details.

(2023); Lukovnikov & Fischer (2025); Bansal et al. (2023); Liu & Liu (2023); Qi et al. (2023); Bar-Tal et al. (2023) or by fine-tuning the base model Xue et al. (2023); Wang et al. (2023b); Zhang et al. (2025a); Zeng et al. (2023). Most related, Seg2Any Zhang et al. (2025b) builds on Flux using training-time attention restriction and region contour maps for boundary precision, but does not support image references.

Reference image guidance. Appearance-transfer methods inject reference features via cross-attention or in-context conditioning (AnyDoor Chen et al. (2024c), MimicBrush Chen et al. (2024b), Insert Anything Song et al. (2025), Ace++ Mao et al. (2025)), while DreamRenderer Zhou et al. (2025a) encodes references into text embeddings via Redux Labs (2024). Crucially, these condition a *single region at a time* and entangle signals globally, whereas *AppearancePointers* routes each cue to its region via appearance pointers and generates multiple heterogeneous regions in one denoising pass.

3 METHOD

Overview. Recent state-of-the-art image generative models Labs (2024); Labs et al. (2025); Esser et al. (2024) are transformer-based and jointly process textual and visual tokens through full self-attention, operating on two streams: a *text stream* encoding the prompt and a *visual stream* encoding conditioning images and generative tokens. Our method generates an image $I \in \mathbb{R}^{H\times W\times 3}$ that satisfies both global and regional conditions. The global condition is provided as a text prompt, while the regional conditions are defined as a set of n pairs $\mathcal{R} = \{(R_i, P_i)\}_{i=1}^{n}$. Here, each $R_i \in \{0,1\}^{H\times W}$ denotes a region mask, and P_i represents the corresponding local prompt, which may consist of tokens coming from text descriptions, reference images, or both. The model is tasked to generate an image where each region corresponding to R_i follows the content P_i, while the rest of the image aligns with the global text prompt (see Figure 2).

While vanilla DiTs possess the ability of ingesting image or text tokens, simply passing additional tokenized information is not enough to inform the models about the localization and composition of the elements in the final image. To this end, we introduce a new type of input token responsible for informing the DiT where and how localized text and image information should be used during generation. We call these ***appearance pointers***. Their role is to ensure correct linking between regions (R_i) and prompts (P_i). In practice, as shown in fig. 2, the *appearance pointers* serve as a conditional input to the DiT model, alongside global and region-specific information — *local image prompts* and

local text prompts complement our *appearance pointers* to provide fine-grained appearance details, while *region contour maps* promote fine-grained alignment with region boundaries.

3.1 Appearance Pointers

We first describe how we construct *appearance pointers*. To align with our objectives, a good regional representation $\mathcal{R}$ should enable multi-modal control while explicitly linking the local prompts P_i to the intended region R_i. To achieve this, we design a *Region Correspondence Transformer* that links and processes all modalities jointly, before passing them to the DiT generative model. We first describe how each region R_i and its corresponding prompt P_i are encoded, and then explain how these representations are aligned to form the *appearance pointers*.

Region and Condition Encoding. Each regional information (R_i, P_i) consists of a binary mask R_i and a local prompt P_i, which may be text, an image, or both. Image prompts are encoded with the VAE image encoder providing image tokens ${}^{I}\mathcal{P}_i$, while text prompts are encoded with the Flux Kontext T5 encoder Labs et al. (2025) providing text tokens ${}^{T}\mathcal{P}_i$. To encode the mask itself, we treat it as an image input to the same VAE encoder. Before encoding, we augment the binary mask with coordinate information to provide explicit spatial grounding. Please see supplement for more details. The coordinate-enhanced mask M_i is then encoded and patchified in the same manner as an image prompt, yielding mask tokens with explicit spatial context.

Region-Prompt Linking. After encoding masks, images, and text to the DiT latent space, our next goal is to *link* masks to their respective condition tokens. This *linking* enables DiTs to *point* to the desired condition signal and helps generate region conditioned images (see fig. 20 for pointer attention visualization). To integrate spatial constraints with semantic content, we introduce a *Region Correspondence Transformer*, $\mathbf{\Phi}_{RC}$ to process tokens from the mask ${}^{R}\mathcal{P}_i$, image ${}^{I}\mathcal{P}_i$, and text ${}^{T}\mathcal{P}_i$ prompts to produce two semantic feature maps, ${}^{I}M_i$ and ${}^{T}M_i$ for each region i:

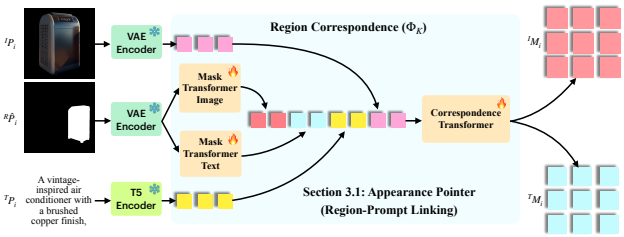

Figure 3: **Region-Prompt Linking.** We generate feature maps targeting both the image and text streams of the DiT backbone; either prompt may be omitted (section 3.1).

$$ {}^{I}M_i,\ {}^{T}M_i := \mathbf{\Phi}_{RC}([\,{}^{R}\hat{\mathcal{P}}_i,\ {}^{I}\mathcal{P}_i,\ {}^{T}\mathcal{P}_i\,]). \tag{1}$$

These maps serve as modular semantic region representations that guide region-conditioned image generation. Specifically, ${}^{I}M_i$ and ${}^{T}M_i$ respectively are targeting the image and text streams of the backbone DiT model, enabling full utilization of its multi-modal conditioning mechanism.

Within $\mathbf{\Phi}_{RC}$, the region tokens undergo initial alignment via two lightweight mask transformer (2 layer self-attention) blocks aligning mask with the corresponding image and text condition. This mask transformer also undergoes token downsampling analogous to U-DiT Tian et al. (2024) for efficiency (see fig. 3). The processed mask tokens, image, and text tokens are concatenated along the sequence dimension and provided to a multi-modal Correspondence Transformer with self-attention blocks (consistent with the Flux MM-DiT attention). These attention blocks use separate query, key, and value projections for each tokens type. In practice, we also pass learnable tokens to account for missing modalities allowing the module to operate flexibly on image+mask, text+mask, or image+text+mask inputs. Crucially, Φ_{RC} operates on each region independently and is diffusion-step independent. This enables us to run region correspondence only once per generation, significantly reducing overhead. Additionally, this design allows the number of regions to vary dynamically during inference.

Region Aggregation. With the regions linked to their corresponding conditions we can condition the diffusion transformer to generate region controlled images. However, naively injecting all region tokens along with their conditional tokens increases the token count drastically and is unfeasible with $\mathcal{O}(N^2)$ complexity for self-attention. Hence, we create *Appearance Pointers* by consolidating information across regions using a *Region Aggregation Transformer* into a single canvas producing Appearance Pointers (see fig. 4).

To preserve spatial structure, we perform this aggregation locally at each patch location. We first collect each semantic region embedding ${}^{T}M_i, {}^{I}M_i \in \mathbb{R}^{N\times C}$ from eq. (1) and stack them along the region dimension to obtain ${}^{I}\mathcal{M}, {}^{T}\mathcal{M} \in \mathbb{R}^{N\times R\times C}$.

Here, N is the number of spatial tokens, R is the number of regions and C is the channel dimension. Next, the region aggregation transformer performs region-wise self-attention for each patch independently, treating R as the sequence length. To consolidate the information, we prepend a learnable `[CLS]` token ($\mathcal{A}$) to the region sequence at each patch. This "depth-wise" processing collapses the multi-region stack into a single, unified semantic canvas that matches the spatial dimensions of the backbone DiT.

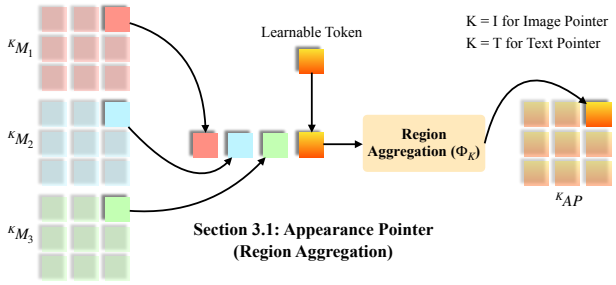

Figure 4: **Appearance pointers generation.** The *Region Aggregation Transformer* creates a concise representation of regional conditions, *Appearance Pointers*.

In practice, we employ two independent aggregation blocks, Φ_A^T and Φ_A^I, to process the text and image streams separately. We extract the updated `[CLS]` tokens to serve as the final appearance pointers:

$$ {}^{T}\mathcal{AP} := \Phi_A^T([{}^{T}A, {}^{T}\mathcal{M}]), \tag{2} $$

$$ {}^{I}\mathcal{AP} := \Phi_A^I([{}^{I}A, {}^{I}\mathcal{M}]). \tag{3} $$

The appearance pointer aggregation module improves identity and coherence resulting better identity preservation and image coherence (Ablation table 5).

3.2 Base DiT Conditioning

Dual Stream Conditioning. We pass the *text* (${}^{T}\mathcal{AP}$) and *image* (${}^{I}\mathcal{AP}$) *appearance pointers* to the image and text streams of the Flux Kontext model, respectively, to guide the generation process. Additionally, we include the individual local image ${}^{I}\mathcal{P}_i$ and text ${}^{T}\mathcal{P}_i$ prompt tokens. Concatenating all tokens yields the conditions for the image and text streams:

$$ X_t := [x_t; {}^{I}\mathcal{AP}; {}^{I}\mathcal{P}_1; \ldots; {}^{I}\mathcal{P}_n] \quad \text{(Image Stream)} \tag{4} $$

$$ c := [{}^{G}\mathcal{P}; {}^{T}\mathcal{AP}; {}^{T}\mathcal{P}_1; \ldots; {}^{T}\mathcal{P}_n] \quad \text{(Text Stream)} \tag{5} $$

where x_t are the noisy tokens. The denoising step of the model is then written as:

$$ x_{t-1} := \text{FLUX}(X_t, c, t). \tag{6} $$

Note that some of the local prompts ${}^{I}\mathcal{P}_i$ and ${}^{T}\mathcal{P}_i$ may be empty.

Region Contour Guidance. The *appearance pointers* effectively link image regions with image and text prompts. However, its token-reduction design of the mask transformer and region aggregation occasionally smooth over fine-grained details. To address this limitation, inspired by DreamRenderer Zhou et al. (2025a) and Seg2Any Zhang et al. (2025b), we aggregate all edges into a single boundary map, which is encoded using the FLUX VAE and provided to the DiT for improved precision (Ablation table 5).

Implementation Details. We train *AppearancePointers* using a flow-matching objective with log-normal time sampling on 8×A100 GPUs for three days. Following OminiControl Tan et al. (2025), we apply LoRA with rank 128 to the newly introduced conditional *Appearance Pointer* tokens and *edge* tokens. All learnable parameters are trained using Prodigy Mishchenko & Defazio (2023) with a learning rate of 1.0. Additional training details and pseudo-code (algorithm 1) are provided in the supplement.

Complexity. In terms of memory, our Region Aggregation and Region Correspondence modules consist of ∼400M parameters – a 3.33% increase over the number of parameters of the base model

and lora is ∼500M parameters. In terms of time, it is important to highlight that Appearance Pointers are computed once for the whole inference procedure rather than per denoising timestep. Moreover, the design we propose with spatial downsampling and independent region processing reduces the attention computation complexity from $\mathcal{O}(T \cdot (RN_{\text{reg}})^2)$ to $\mathcal{O}(R(N_{\text{reg}}/k)^2)$, where k is the downsampling factor, R is the number of regions, N_{reg} is the number of tokens per region, ant T is the number or regions. This ensures negligible overhead relative to the base generation process maintaining high inference efficiency.

4 AppearancePointers-*37K* Data Generation

Existing personalization and object-insertion datasets Tan et al. (2025); Saha et al. (2025); Zhang et al. (2025b) lack fine-grained region captions. We instead generate a fine-grained, region-level editing dataset enabling controlled variations in object orientation, texture, and material, with coverage across small, medium, and large objects. We generate the dataset in three stages, described below and visualized in fig. 5.

Scene Captioning. We begin by categorizing everyday objects into three scale groups: small (graspable objects), medium (indoor items such as beds, tables, and chairs), and large (vehicles and infrastructure such as cars, buses, cranes, and buildings). For each scene, we sample a variable number of objects from these categories and provide a coarse scene context description (*e.g.*, "kitchen", "living room") to an LLM. Using Qwen 3 Yang et al. (2025a), we generate: (i) a global scene description, and (ii) object-level descriptions that include visual attributes, a material description, and a dedicated description for accurate object grounding.

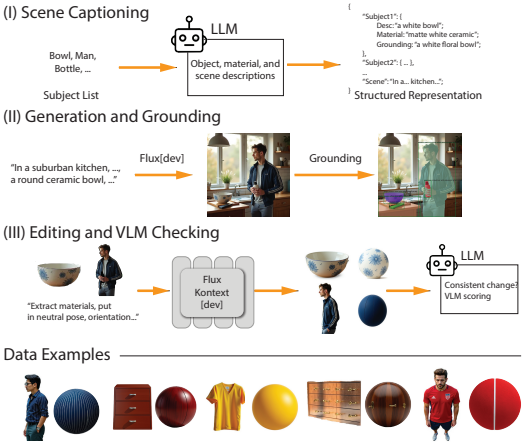

Figure 5: **Overview of the dataset creation pipeline.** (I) We generate diverse scenes by sampling objects of varying scales and prompting an LLM to produce scene- and object-level descriptions. (II) Images are synthesized using Flux.1 Dev and grounded with Grounded SAM. (III) For each object, we produce pose and material edits and filter them using a VLM.

Generation and Grounding. The structured scene prompt is provided to Flux.1 Dev Labs (2024) to generate an image. We then segment each described object using Grounded SAM Ren et al. (2024), guided by dedicated descriptors. This yields region masks linking each generated region to its text reference.

Editing and VLM Checking. Following SIGMA-Gen Saha et al. (2025), we extract each grounded object and generate two types of controlled edits: (i) pose perturbations, and (ii) material and texture variations, produced using Flux Kontext. Each edited crop is evaluated by an InternVL Chen et al. (2024d) vision–language model, verifying whether the edits remain consistent with the original descriptions through VLM-based scoring. Full pipeline details and prompting templates are provided in the supplementary material.

5 Experiments & Results

5.1 Quantitative Results

Dataset. Since no existing benchmark offers both text- and image-conditioned regions in one scene, we benchmark on our *Appearance Pointers-37K* (section 4) for region, text, and image conditioned generation, and validate real-world transfer on SACap-eval (table 4).

Baselines. As prior works do not support conditioning regions on both text and images, we benchmark *AppearancePointers* against unimodal regional models: *InstanceDiffusion* Wang et al. (2024), *DreamRenderer* Zhou et al. (2025a), and *Seg2Any* Zhang et al. (2025b), — for text+region-to-image generation (table 2 and fig. 8). To evaluate image+region-to-image generation, we also compare

Table 2: Quantitative comparison on AppearancePointers-37K (textual region descriptions). Best result in gold, second best in silver. Higher is better for all metrics.

	Region				Global	
	CLIP-I↑	CLIP-T↑	DINO-I↑	MIoU↑	CLIP-T↑	CLIP-IQA↑
InstanceDiffusion Wang et al. (2024)	86.39	27.54	35.02	41.04	26.70	93.37
DreamRenderer Zhou et al. (2025a)	87.75	28.03	44.20	33.84	28.43	93.24
Seg2Any Zhang et al. (2025b)	89.11	27.67	50.05	36.37	29.30	94.59
Ours	90.40	27.24	56.09	40.35	28.93	95.02

AppearancePointers against the prior state-of-the-art method, *MS-Diffusion* Wang et al. (2025b) and DreamRenderer* Zhou et al. (2025a) (table 3 and fig. 6).

We also evaluate an editing variant of *AppearancePointers* that preserves background regions of an input image while performing targeted modifications. For further implementation details, additional examples and comparison against InsertAnything Song et al. (2025), we refer the reader to the supplementary material. For fairness, all baselines use their released weights and we report distribution-neutral metrics (excluding FID/KID).

Metrics. Our benchmark generates 500 images under text-only and image-only regional conditions, each with a global prompt and ∼5 regions; the image-only setting adds novel-view object references. We report two *global* metrics — **CLIP-IQA** Wang et al. (2023a) (visual quality) and global **CLIP-T** (alignment to the global prompt) — and four *region* metrics — **DINO-I** and **CLIP-I** (region fidelity to the reference), region **CLIP-T** (region–prompt alignment), and class-agnostic **MIoU** Zhang et al. (2025b) (shape adherence). Full metric definitions are given in section F.

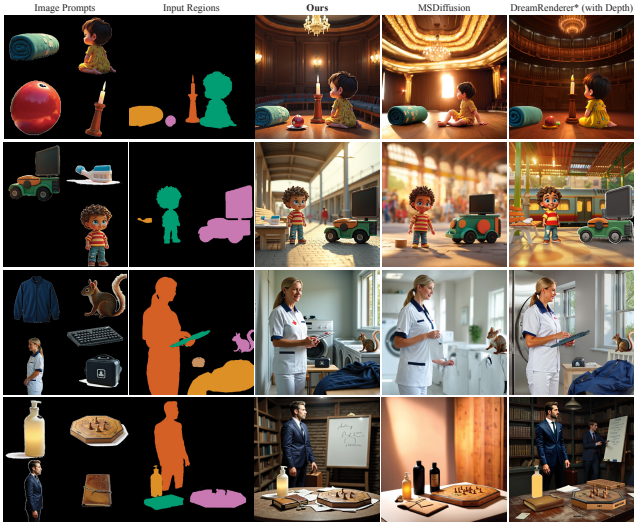

Figure 6: **Qualitative comparison on region-conditioned generation from image prompts.** Each example provides object reference images **(left)** and user-defined region masks **(second column)**, specifying where each object should appear in the final scene. We compare against MSDiffusion, a recent multi-subject generation method & DreamRenderer* that has depth supervision. Across varied scenarios, our results show more accurate object identity and more faithful spatial placement, while MSDiffusion & DreamRenderer often alter appearance or places objects inconsistently.

Appearance-37K evaluation. Table 2 and Table 3 compare our method against prior works for generation with text-conditioned regions and image-conditioned regions, respectively. Qualitative comparisons are in figs. 6 and 8, with extensive galleries (material, image, and text conditions; in-the-wild editing; layout generation) in figs. 11 to 15. As shown in Table 2, on text-conditioned region generation, our method achieves the highest scores in global image quality (CLIP-IQA: 95.02), region fidelity (CLIP-I: 90.40), and semantic alignment (DINO-I: 56.09) for text-region-to-image synthesis. We are also competitive in MIoU (40.35), indicating strong adherence to region masks. In contrast, Seg2Any and DreamRenderer often fail to strictly adhere to the specified region masks, which leads to lower MIoU scores despite competitive CLIP-T performance. For the image-region control benchmark (Table 3), our method outperforms both MS-Diffusion and DreamRenderer*, achieving a CLIP-I score of 93.29, MIoU of 40.97, and DINO-I score of 69.31, demonstrating higher region fidelity, identity, and boundary adherence.

Real-world transfer: We test zero-shot transfer to the real-world SACap-eval benchmark Zhang et al. (2025b), despite *AppearancePointers* being trained only on synthetic data (table 4); for fairness with Seg2Any, which blurs faces, we exclude human subjects and sample 200 images. *AppearancePointers* transfers strongly, leading on aesthetics (CLIP-IQA), while Seg2Any, trained in-distribution on SACap-1M, retains an edge on region–text adherence; fine-tuning on only 2.3% real data narrows this gap and leads on global alignment. See section G for insertion comparisons, baseline settings,

Table 3: Quantitative comparison on AppearancePointers-37K (image region descriptions). Best results in gold.

	Region				Global	
	CLIP-I↑	CLIP-T↑	DINO-I↑	MIoU↑	CLIP-T↑	CLIP-IQA↑
MSDiffusion Wang et al. (2025b)	89.66	24.89	45.61	28.86	31.08	74.85
DreamRenderer* Zhou et al. (2025a)	92.08	27.37	64.20	40.11	30.02	89.56
Ours	93.29	27.25	69.31	40.97	29.62	95.57

real-world qualitative results (fig. 17), and a comparison with Nano-Banana, which does not follow the region layout even when given the masks (fig. 18).

Table 4: Quantitative comparison on SACap-eval dataset (real-world images). Our work is competitive even when trained entirely on synthetic data with better aesthetics and image-text adherance (Zero-Shot setting). Minimal fine-tuning with only **2.3%** of real-data improves region text adherence significantly. AppearancePointer is the only model that supports both image and text as regional guidance with the same weights (highlighted in red). Best results in gold, second best in silver.

	Training Data	Region				Global	
		CLIP-I↑	CLIP-T↑	DINO-I↑	MIoU↑	CLIP-T↑	CLIP-IQA↑
Task: Region+Text to Image							
Seg2Any	SACap-1M	×	26.24	×	64.24	32.30	86.07
Ours (Zero-Shot)	AP-37K	×	25.57	×	60.14	32.17	89.27
Ours (2.3% Fine-Tune)	AP-37K + SACap-23K	×	25.78	×	61.72	32.99	87.92
Task: Region+Image to Image							
MS-Diffusion (Bbox)	Private-3M	87.81	×	46.86	53.53	31.95	74.91
Ours (Zero-Shot, Bbox)	AP-37K	93.92	×	72.16	59.64	32.78	86.23
Ours (Zero-Shot, Fine-Grained)	AP-37K	94.09	×	72.63	62.46	33.02	86.27

Table 5: **Ablation on image-conditioned generation.** Each row removes one component of the *Appearance Pointer* model; see section 5.1 for component-wise analysis.

	Region				Global	
	CLIP-I↑	CLIP-T↑	DINO-I↑	MIoU↑	CLIP-T↑	CLIP-IQA↑
w/o Region Aggregation section 3.1	89.93	27.35	54.47	41.49	28.45	93.48
w/o Region Contour Guidance section 3.2	85.57	23.89	41.62	35.81	28.89	94.47
w/o Position ID Resampling Multi-Region Mimic section B.2	93.01	27.2	67.46	40.75	29.47	95.34
w/o Appearance Pointer Mask section B.3	93.23	27.25	68.93	41.09	29.60	95.83
***AppearancePointers* (Ours)**	93.29	27.25	69.31	40.97	29.62	95.57

Ablations: Table 5 presents ablations of our *AppearancePointers* method for the image-conditioned prompt setting. We use the image-conditioned prompt setting to evaluate the effectiveness of our method in preserving identity. We first remove the region aggregation (section 3.1), and instead of *Appearance Pointers* ${}^{T}\mathcal{AP}$ and ${}^{I}\mathcal{AP}$, directly use ${}^{T}\mathcal{M}$ and ${}^{I}\mathcal{M}$ to condition the DiT. Region Aggregation improves subject identity (from DINO-I 54.47 to 69.31) and image quality (93.48 to 95.57). The Region Contour improves the MIoU from 35.81 to 40.97 improving region precision. Omitting the appearance pointer mask (section B.3) reduces spatial disentanglement and weakens region alignment, while disabling position ID resampling (section B.2) degrades identity and region consistency, most notably in multi-region scenes. Identity (CLIP-I) and global quality stay stable as the region count grows from 1 to 9 and as masks coarsen from fine to bounding-box to elliptical, while region-adherence metrics (DINO-I, MIoU) decline gradually rather than collapsing (tables 8 and 9). See supplement for more details.

Conclusion. We introduced appearance pointers, a lightweight and modality-agnostic mechanism that enables Diffusion Transformers to interpret and apply region-specific user intent from text or image inputs. Our region correspondence network and spatial aggregation module combine multiple regional descriptions within a single denoising process, and we introduce a synthetic multi-region dataset for training and evaluation. When compared to modality-specific approaches, our model reaches or surpasses the state-of-the-art in most of the metrics, despite being trained for multiple modalities. These findings demonstrate that appearance pointers provide a simple, extensible, and effective interface for precise multimodal guidance, enabling generative models to better realize not only *what* users intend to create, but *where* and *how* the content should appear.

Limitations. Our method can overlook fine regions, struggles with fine details such as human facial identity, and loses region adherence beyond $\sim$10 regions; see the supplement for failure cases.

AI USE STATEMENT

We used LLMs and VLMs as components of our synthetic dataset generation pipeline for *AppearancePointers*-37K: Qwen 3 produces the scene and object-level captions, Flux.1 Dev is used to synthesize the images, Grounded SAM performs the region grounding, Flux Kontext produces the pose and material edits, and InternVL scores the edits for consistency filtering (details in the supplement). We additionally used AI-based assistants to help format and polish the writing (grammar, phrasing, and conciseness) and to assist with literature search. All research contributions that include the method, dataset design, experiments, and analysis are our own, and we verified all cited references and take full responsibility for the final content of the paper, including any text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

Our method generates and edits images and, like other generative models, could be misused to produce misleading or infringing content; we encourage responsible use and appropriate safeguards when deploying such systems. Our training data is synthetically generated, and for the real-world evaluation we follow prior work in excluding human subjects. We do not collect or use private or personally identifiable data.

REPRODUCIBILITY STATEMENT

We provide extensive implementation details in the supplement, including the training setup, region-prompt sampling scheme, RoPE positional-ID design, and architecture specifications, together with pseudocode for constructing the appearance pointers (algorithm 1). We further describe the full dataset generation pipeline and include the exact LLM and VLM prompting templates used to generate and verify the data (listing 3). Combined with the method description in section 3, these details are intended to enable reproduction of our approach. We will release open-sourced code & data upon publication.

REFERENCES

Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Qinsheng Zhang, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, et al. ediff-i: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022.

Arpit Bansal, Hong-Min Chu, Avi Schwarzschild, Soumyadip Sengupta, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Universal guidance for diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 843–852, 2023.

Omer Bar-Tal, Lior Yariv, Yaron Lipman, and Tali Dekel. Multidiffusion: Fusing diffusion paths for controlled image generation. *Proceedings of Machine Learning Research*, 2023.

Pu Cao, Feng Zhou, Qing Song, and Lu Yang. Controllable generation with text-to-image diffusion models: A survey. *arXiv preprint arXiv:2403.04279*, 2024.

Minghao Chen, Iro Laina, and Andrea Vedaldi. Training-free layout control with cross-attention guidance. In *2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2024a.

Xi Chen, Yutong Feng, Mengting Chen, Yiyang Wang, Shilong Zhang, Yu Liu, Yujun Shen, and Hengshuang Zhao. Zero-shot image editing with reference imitation. *Advances in Neural Information Processing Systems*, 37, 2024b.

Xi Chen, Lianghua Huang, Yu Liu, Yujun Shen, Deli Zhao, and Hengshuang Zhao. Anydoor: Zero-shot object-level image customization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6593–6602, 2024c.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning

for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 24185–24198, 2024d.

Guillaume Couairon, Marlene Careil, Matthieu Cord, Stéphane Lathuiliere, and Jakob Verbeek. Zero-shot spatial layout conditioning for text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

Xiao Cui, Qi Sun, Min Wang, Li Li, Wengang Zhou, and Houqiang Li. Layoutenc: Leveraging enhanced layout representations for transformer-based complex scene synthesis. *ACM Transactions on Multimedia Computing, Communications and Applications*, 21(4):1–21, 2025.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Proceedings of the 41st International Conference on Machine Learning*, 2024. URL https://proceedings.mlr.press/v235/esser24a.html.

Cusuh Ham, James Hays, Jingwan Lu, Krishna Kumar Singh, Zhifei Zhang, and Tobias Hinz. Modulating pretrained diffusion models for multimodal image synthesis. In *ACM SIGGRAPH 2023 Conference Proceedings*, 2023.

Yutong He, Ruslan Salakhutdinov, and J Zico Kolter. Localized text-to-image generation for free via cross attention control. *arXiv preprint arXiv:2306.14636*, 2023.

Chengyou Jia, Minnan Luo, Zhuohang Dang, Guang Dai, Xiaojun Chang, Mengmeng Wang, and Jingdong Wang. Ssmg: Spatial-semantic map guided diffusion model for free-form layout-to-image generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 2024.

Rui Jiang, Guang-Cong Zheng, Teng Li, Tian-Rui Yang, Jing-Dong Wang, and Xi Li. A survey of multimodal controllable diffusion models. *Journal of Computer Science and Technology*, 39(3): 509–541, 2024.

Yunji Kim, Jiyoung Lee, Jin-Hwa Kim, Jung-Woo Ha, and Jun-Yan Zhu. Dense text-to-image generation with attention modulation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7701–7711, 2023.

Black Forest Labs. Flux.1: Official inference repository for flux open-weight models. https://github.com/black-forest-labs/flux, 2024.

Black Forest Labs, Stephen Batifol, A. Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, Sumith Kulal, Kyle Lacey, Yam Levi, Cheng Li, Dominik Lorenz, Jonas Muller, Dustin Podell, Robin Rombach, Harry Saini, Axel Sauer, and Luke Smith. Flux.1 kontext: Flow matching for in-context image generation and editing in latent space. *ArXiv*, abs/2506.15742, 2025. URL https://api.semanticscholar.org/CorpusID:279464475.

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

Zejian Li, Jingyu Wu, Immanuel Koh, Yongchuan Tang, and Lingyun Sun. Image synthesis from layout with locality-aware mask adaption. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

Chang Liu and Dong Liu. Late-constraint diffusion guidance for controllable image synthesis. *arXiv preprint arXiv:2305.11520*, 2023.

Jiaqi Liu, Tao Huang, and Chang Xu. Training-free composite scene generation for layout-to-image synthesis. In *European Conference on Computer Vision*, pp. 37–53. Springer, 2024.

Denis Lukovnikov and Asja Fischer. Enabling controlnet to follow localized descriptions using cross-attention control. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 310–327. Springer, 2025.

Chaojie Mao, Jingfeng Zhang, Yulin Pan, Zeyinzi Jiang, Zhen Han, Yu Liu, and Jingren Zhou. Ace++: Instruction-based image creation and editing via context-aware content filling. *arXiv preprint arXiv:2501.02487*, 2025.

Jiafeng Mao and Xueting Wang. Training-free location-aware text-to-image synthesis. In *2023 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2023.

Jiafeng Mao, Xueting Wang, and Kiyoharu Aizawa. Guided image synthesis via initial image editing in diffusion model. In *Proceedings of the 31st ACM International Conference on Multimedia*, 2023a.

Jiafeng Mao, Xueting Wang, and Kiyoharu Aizawa. Semantic-driven initial image construction for guided image synthesis in diffusion model. *arXiv preprint arXiv:2312.08872*, 2023b.

Konstantin Mishchenko and Aaron Defazio. Prodigy: An expeditiously adaptive parameter-free learner. *ArXiv*, abs/2306.06101, 2023. URL https://api.semanticscholar.org/CorpusID:259129271.

Sicheng Mo, Fangzhou Mu, Kuan Heng Lin, Yanli Liu, Bochen Guan, Yin Li, and Bolei Zhou. Freecontrol: Training-free spatial control of any text-to-image diffusion model with any condition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, Ying Shan, and Xiaohu Qie. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. *arXiv preprint arXiv:2302.08453*, 2023.

Yunfang Niu, Lingxiang Wu, Yufeng Zhang, Yousong Zhu, Guibo Zhu, and Jinqiao Wang. Multi-model style-aware diffusion learning for semantic image synthesis. *ACM Transactions on Multimedia Computing, Communications and Applications*, 20(11), 2024.

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.

Bohao Peng, Jian Wang, Yuechen Zhang, Wenbo Li, Ming-Chang Yang, and Jiaya Jia. Controlnext: Powerful and efficient control for image and video generation. *arXiv preprint arXiv:2408.06070*, November 2024.

Quynh Phung, Songwei Ge, and Jia-Bin Huang. Grounded text-to-image synthesis with attention refocusing. *arXiv preprint arXiv:2306.05427*, 2023.

Zipeng Qi, Guoxi Huang, Zebin Huang, Qin Guo, Jinwen Chen, Junyu Han, Jian Wang, Gang Zhang, Lufei Liu, Errui Ding, et al. Layered rendering diffusion model for zero-shot guided image synthesis. *arXiv preprint arXiv:2311.18435*, 2023.

Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024.

Oindrila Saha, Vojtech Krs, Radomir Mech, Subhransu Maji, Kevin Blackburn-Matzen, and Matheus Gadelha. Sigma-gen: Structure and identity guided multi-subject assembly for image generation. *arXiv preprint arXiv:2510.06469*, 2025.

Takahiro Shirakawa and Seiichi Uchida. Noisecollage: A layout-aware text-to-image diffusion model based on noise cropping and merging. *arXiv preprint arXiv:2403.03485*, 2024.

Xincheng Shuai, Henghui Ding, Xingjun Ma, Rongcheng Tu, Yu-Gang Jiang, and Dacheng Tao. A survey of multimodal-guided image editing with text-to-image diffusion models. *arXiv preprint arXiv:2406.14555*, 2024.

Wensong Song, Hong Jiang, Zongxing Yang, Ruijie Quan, and Yi Yang. Insert anything: Image insertion via in-context editing in dit. *arXiv preprint arXiv:2504.15009*, 2025.

Wenqiang Sun, Teng Li, Zehong Lin, and Jun Zhang. Spatial-aware latent initialization for controllable image generation. *arXiv preprint arXiv:2401.16157*, 2024.

Ashkan Taghipour, Morteza Ghahremani, Mohammed Bennamoun, Aref Miri Rekavandi, Hamid Laga, and Farid Boussaid. Box it to bind it: Unified layout control and attribute binding in text-to-image diffusion models. *IEEE transactions on multimedia*, 2025.

Zhenxiong Tan, Songhua Liu, Xingyi Yang, Qiaochu Xue, and Xinchao Wang. Ominicontrol: Minimal and universal control for diffusion transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 14940–14950, 2025.

Yuchuan Tian, Zhijun Tu, Hanting Chen, Jie Hu, Chao Xu, and Yunhe Wang. U-dits: Downsample tokens in u-shaped diffusion transformers. In *Advances in Neural Information Processing Systems (NeurIPS) 37*, 2024.

Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Herv'e J'egou. Going deeper with image transformers. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 32–42, 2021. URL https://api.semanticscholar.org/CorpusID:232428161.

Jia Wang, Jie Hu, Xiaoqi Ma, Hanghang Ma, Xiaoming Wei, and Enhua Wu. Image editing with diffusion models: A survey. *arXiv preprint arXiv:2504.13226*, 2025a.

Jianyi Wang, Kelvin CK Chan, and Chen Change Loy. Exploring clip for assessing the look and feel of images. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 2555–2563, 2023a.

Weilun Wang, Jianmin Bao, Wengang Zhou, Dongdong Chen, Dong Chen, Lu Yuan, and Houqiang Li. Semantic image synthesis via diffusion models. *arXiv preprint arXiv:2207.00050*, 2022.

Xierui Wang, Siming Fu, Qihan Huang, Wanggui He, and Hao Jiang. MS-diffusion: Multi-subject zero-shot image personalization with layout guidance. In *The Thirteenth International Conference on Learning Representations*, 2025b. URL https://openreview.net/forum?id=PJqP0wyQek.

Xudong Wang, Trevor Darrell, Sai Saketh Rambhatla, Rohit Girdhar, and Ishan Misra. Instancediffusion: Instance-level control for image generation, 2024.

Yibin Wang, Weizhong Zhang, Jianwei Zheng, and Cheng Jin. Enhancing object coherence in layout-to-image synthesis. *arXiv preprint arXiv:2311.10522*, 2023b.

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. *arXiv preprint arXiv:2502.13081*, 2025.

Jiayu Xiao, Liang Li, Henglei Lv, Shuhui Wang, and Qingming Huang. R&b: Region and boundary aware zero-shot grounded text-to-image generation. *arXiv preprint arXiv:2310.08872*, 2023.

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

Han Xue, Zhiwu Huang, Qianru Sun, Li Song, and Wenjun Zhang. Freestyle layout-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025a.

Binbin Yang, Yi Luo, Ziliang Chen, Guangrun Wang, Xiaodan Liang, and Liang Lin. Law-diffusion: Complex scene generation by diffusion with layouts. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

Pengfei Yang, Ngai-Man Cheung, and Xinda Ma. Text to image generation and editing: A survey. *arXiv preprint arXiv:2505.02527*, 2025b.

Yu Zeng, Zhe Lin, Jianming Zhang, Qing Liu, John Collomosse, Jason Kuen, and Vishal M Patel. Scenecomposer: Any-level semantic image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Zheyuan Zhan, Defang Chen, Jian-Ping Mei, Zhenghe Zhao, Jiawei Chen, Chun Chen, Siwei Lyu, and Can Wang. Conditional image synthesis with diffusion models: A survey. *arXiv preprint arXiv:2409.19365*, 2024.

Hong Zhang, Zhongjie Duan, Xingjun Wang, Yingda Chen, and Yu Zhang. Eligen: Entity-level controlled image generation with regional attention. *arXiv preprint arXiv:2501.01097*, 2025a.

Hui Zhang, Sheng Wang, Jiacheng Li, Zuxuan Wu, et al. Seg2any: Open-set segmentation-mask-to-image generation with precise shape and semantic control. *arXiv preprint arXiv:2506.00596*, 2025b.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *ICCV*, 2023.

Peiang Zhao, Han Li, Ruiyang Jin, and S Kevin Zhou. Loco: Locally constrained training-free layout-to-image synthesis. *arXiv preprint arXiv:2311.12342*, 2023.

Shihao Zhao, Dongdong Chen, Yen-Chun Chen, Jianmin Bao, Shaozhe Hao, Lu Yuan, and Kwan-Yee K Wong. Uni-controlnet: All-in-one control to text-to-image diffusion models. *Advances in Neural Information Processing Systems*, 36, 2024.

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

Zirui Zheng, Takashi Isobe, Tong Shen, Xu Jia, Jianbin Zhao, Xiaomin Li, Mengmeng Ge, Baolu Li, Qinghe Wang, Dong Li, et al. Layout-conditioned autoregressive text-to-image generation via structured masking. *arXiv preprint arXiv:2509.12046*, 2025.

Dewei Zhou, Mingwei Li, Zongxin Yang, and Yi Yang. Dreamrenderer: Taming multi-instance attribute control in large-scale text-to-image models. *arXiv preprint arXiv:2503.12885*, 2025a.

Dewei Zhou, Ji Xie, Zongxin Yang, and Yi Yang. 3dis-flux: simple and efficient multi-instance generation with dit rendering. *arXiv preprint arXiv:2501.05131*, 2025b.

APPENDIX CONTENTS

A Extended Related Works **15**

B Implementation Details & Pseudocode **16**

B.1 Training Setting 16

B.2 Conditions' RoPE Positional IDs 17

B.3 Appearance Pointer Mask 17

B.4 Architecture 18

C Multi-Subject Insertion (Editing) **19**

D Ablations **19**

D.1 Appearance Pointer Pseudocode 20

E Qualitative Results **20**

F Evaluation Metrics **29**

G Baseline Comparison Setting **29**

H Limitation & Discussion with Future Works **30**

I Appearance Pointer Attention Map Analysis **32**

J Additional Dataset Details **32**

J.1 Dataset VLM Check 32

J.1.1 Novel View Verification 32

J.1.2 Material/Texture Verification 32

J.2 Prompting 33

A EXTENDED RELATED WORKS

For detailed discussions on multimodal controllable diffusion models, we refer the reader to recent surveys Cao et al. (2024); Jiang et al. (2024); Shuai et al. (2024); Zhan et al. (2024); Wang et al. (2025a); Yang et al. (2025b). Earlier methods operated on segmentation maps, employing specialized architectures Niu et al. (2024); Wang et al. (2022); Ham et al. (2023) or general conditioning frameworks Zhang et al. (2023); Mou et al. (2023); Peng et al. (2024); Zhao et al. (2024); Mo et al. (2024); Lukovnikov & Fischer (2025), with some extending to bounding box layouts Li et al. (2021); Yang et al. (2023). There are several relevant capabilities on localized text/image conditioned image generation. One of our main goals while developing *AppearancePointers* is to enable a variety of workflows with the same framework without resorting to a big and complicated architecture. A summary of the features supported by *AppearancePointers* and a comparison with state-of-the-art methods can be found in Table 1. Below we discuss previous literature on region-based controls for image synthesis.

Bounding boxes. Early works explored bounding-box-guided text-to-image generation using object names Mao & Wang (2023); Xie et al. (2023) or descriptive prompts Shirakawa & Uchida (2024); Jia et al. (2024), often with a global scene-level prompt Li et al. (2023); Shirakawa & Uchida (2024);

Xie et al. (2023); Xiao et al. (2023); Zhao et al. (2023); Jia et al. (2024), though these lack precise shape controllability. *Training-free* methods incorporate spatial or semantic guidance at inference time via noise initialization Mao et al. (2023b;a); Sun et al. (2024), cross-attention manipulation Mao & Wang (2023); Chen et al. (2024a); Taghipour et al. (2025); Balaji et al. (2022); Zhou et al. (2025a), and sometimes self-attention Liu et al. (2024); Zhou et al. (2025a;b). Taghipour et al. Taghipour et al. (2025) use KL divergence for stronger attribute–region association. An alternative strategy estimates object-specific noises independently and fuses them via cross-attention Shirakawa & Uchida (2024). Several methods apply gradient-based guidance w.r.t. cross-attention alignment Xie et al. (2023); Chen et al. (2024a); Sun et al. (2024), cross- and self-attention consistency Zhao et al. (2023), or alignment augmented with a boundary-aware loss Xiao et al. (2023). *Training-based* approaches introduce lightweight modules to fine-tune the backbone. GLIGEN Li et al. (2023) injects gated self-attention layers conditioned on grounding tokens. SSMG Jia et al. (2024) uses a ControlNet-like branch encoding spatial and semantic features, with attention modeling inter-object relations. LayoutDiffusion Zheng et al. (2023) proposes a layout encoding and fusion module with cross-attention alignment in an end-to-end diffusion framework. Several works also target auto-regressive generation Cui et al. (2025); Zheng et al. (2025). While our method supports both coarse regions (like bounding boxes) and also has the capability of following precise regions. Additionally, it can be employed in both image generation and editing workflows.

Semantic layouts. Many training-free methods manipulate cross-attention to strengthen concept influence within regions He et al. (2023); Kim et al. (2023), optionally restricting self-attention across segments Kim et al. (2023). Gradient-based guidance updates latents via text–latent cross-attention alignment Couairon et al. (2023); Phung et al. (2023); Lukovnikov & Fischer (2025) and sometimes self-attention Phung et al. (2023), image-space losses projected to latent space Bansal et al. (2023), or a lightweight aligner module Liu & Liu (2023). One approach Qi et al. (2023) generates objects independently in early denoising steps before merging, while MultiDiffusion Bar-Tal et al. (2023) optimizes the image at each noise step for global consistency. Training-based methods integrate attention manipulation to fine-tune the base model Xue et al. (2023); Wang et al. (2023b); Zhang et al. (2025a), or inject text- and mask-conditioned features via convolutional layers Zeng et al. (2023). Most closely related, Seg2Any Zhang et al. (2025b) builds on Flux with attention restriction during training and region contour maps for boundary precision, but does not support image references. In contrast, *AppearancePointers* unifies text and image regional conditioning within the same framework, allowing heterogeneous appearance cues to be combined across regions in a single denoising pass.

Reference image guidance. For a comprehensive overview of appearance transfer, we refer to Wei et al. (2025). DreamRenderer Zhou et al. (2025a) focuses on text-based region control and encodes reference images into text embeddings via Redux Labs (2024). AnyDoor Chen et al. (2024c) uses DINOv2 Oquab et al. (2023) features injected via cross-attention alongside ControlNet Zhang et al. (2023)-style conditioning for masks and structure. MimicBrush Chen et al. (2024b) employs dual U-Nets, injecting reference attention keys and values into an imitative U-Net. Insert Anything Song et al. (2025) builds on Flux with reference image, target region, and text conditioning; Ace++ Mao et al. (2025) uses channel-wise concatenation but is outperformed by Song et al. (2025). Unlike our work, these approaches support only *a single region at a time*, whereas *AppearancePointers* generates multiple regions (described by images or text) in a single diffusion pass. Furthermore, while these methods entangle conditioning signals globally or process regions sequentially, *AppearancePointers* routes each appearance cue to its correct spatial location via appearance pointer tokens, enabling precise and modular multimodal control without architectural redundancy.

B Implementation Details & Pseudocode

B.1 Training Setting

We train with up to 7 regions (including background) at each training iteration, where one of the regions can be a background region. During training, we randomly replace the global prompt with a background only global prompt that does not describe the foreground objects 50% of the time. Using background prompt as the global prompt forces region modules to attend more to the region prompts and allows the model to work robustly in scenarios where no global prompt is provided.

As described in section 4 in the main manuscript, we perform VLM checking for generated (i) pose variation and (ii) material and texture variation. We use the generated result for training only when InternVL Chen et al. (2024d) consistency scores exceed 0.84 for pose variation and 0.79 for texture/material variation. The equation below provides the sampling probability p of a region prompt being a subject image from the same viewpoint as in the groundtruth target, a subject image from a novel viewpoint, a material reference image, a text, or their combination:

$$
{}^{I}\mathcal{P}, {}^{T}\mathcal{P} = \begin{cases} p \in [0, 0.4], \begin{cases} p \in [0.0, 0.2] \text{ \& (if consistent novel view)} \\ \quad \text{novel view image } \textbf{(I + T)} \\ p \in (0.2, 0.4] \text{ \& (if consistent material)} \\ \quad \text{material image } \textbf{(I + T)} \\ \text{else, image crop (no text prompt) } \textbf{(I)} \end{cases} \\ p \in (0.4, 0.6], \text{image crop (no text prompt) } \textbf{(I)} \\ p \in (0.6, 1.0], \text{text only prompt } \textbf{(T)}, \end{cases} \tag{7}
$$

where **(I)** denotes *image* prompt and **(T)** denotes *text* prompt with cases of **(I+T)** denotes both image and text prompts. During training, we center each reference subject in the prompt image and apply a black background mask before passing it to our model. Additionally, we sample coarse masks 30% of the time where the coarse mask can be a bounding box or an ellipse. Additionally, we provide the category name for the material prompt as the material alone does not denote the type of subject to be inserted.

B.2 Conditions' RoPE Positional IDs

Following Tan et al. (2025); Saha et al. (2025), we assign distinct, non-overlapping positional embeddings to each conditioning image and to the text-prompt tokens. For an image prompt, the positional IDs are defined as

$$[1,\ R(i) \cdot H + h,\ w],$$

and for text tokens as

$$[0,\ R(i) \cdot H + \text{mean}({}^{T}h),\ 0],$$

where $R(i)$ is the integer region identifier and H is the height dimension of the positional-embedding grid for the generated image, and h, w are the row and column coordinates of the token within the region. Here, mean$({}^{T}h)$ is the mean height of the region prompt linking tokens $({}^{T}\mathcal{M}_i)$ corresponding to the regional text prompt ${}^{T}\mathcal{P}_i$. The *appearance pointer* tokens use positional IDs: $[1,\ h,\ w]$ for the image appearance pointer, and $[0,\ h,\ w]$ for the text appearance pointer.

ID Resampling. To mimic a larger number of regions in a small-region training setting, we multiply the region position ID by a random integer multiplier. This allows the model to experience a wider range of positional IDs, which is necessary for scaling up at inference time.

With this approach, the image prompt token positional ID becomes

$$[1,\ R(i) \cdot m \cdot H + h,\ w],$$

and the regional text prompt token positional ID becomes

$$[0,\ R(i) \cdot m \cdot H,\ 0],$$

where m is a randomly chosen integer in the range $m \in [1, 6]$.

B.3 Appearance Pointer Mask

In the presence of multiple regions, the appearance region pointer aggregation may find it difficult to select the most relevant features. We assist the appearance pointer aggregation by masking regions of interest using the input region mask. Before aggregation all region conditions to obtain the image $({}^{I}\mathcal{AP})$ and text $({}^{T}\mathcal{AP})$ appearance pointers (defined in the main document), we re-purpose the input regional masks $({}^{R}\mathcal{P})$ to ensure non-zero features only in the prompts designated spatial regions. This is implemented as follows:

$${}^{T}\mathcal{AP} := \Phi_A^T([{}^{T}A, {}^{T}\mathcal{M} \odot {}^{R}\mathcal{P}]), \tag{8}$$

$${}^{I}\mathcal{AP} := \Phi_A^I([{}^{I}A, {}^{I}\mathcal{M} \odot {}^{R}\mathcal{P}]). \tag{9}$$



Figure 7: **Multi-Subject Insertion.** we perform multi subject insertion with *AppearancePointers* and benchmark against iterative InsertAnything. InsertAnything tends to degrade image results after repeated subject insertion and **deforms objects such as the white bottle in row 1, the table and leaves in row 2**.

The region-linked features ${}^{T}\mathcal{M} \in \mathbb{R}^{R\times C\times H_t\times W_t}$ and ${}^{I}\mathcal{M} \in \mathbb{R}^{R\times C\times H_i\times W_i}$, the *Region Aggregation Transformer* networks Φ_A^I and Φ_A^T, and the learnable aggregation tokens ${}^{I}A$ and ${}^{T}A$ are defined in the main document. We omit the downsample function used to resize regional masks ${}^{R}\mathcal{P}$ for clarity. $\odot$ represents the Hadamard Product.

B.4 Architecture

The region correspondence block is a six layer multimodal transformer. Each transformer block within region correspondence has separate query, key, and value projections for text, image, and mask tokens. We also have a modulation block that takes the region's clip feature vector to modulate the region similar to attention in FLUX Kontext dev. The tokens are then passed through an MLP with skip connection. Our hidden dimension is 768 and we perform a bottleneck operation that converts from the high dimensional FLUX embedding (D=3072) to 768. This makes our approach lightweight and fast without any perceivable change in performance. The region aggregation block is constructed in a similar way with the only difference that we don't have modulation and instead skip connections with layer scale Touvron et al. (2021). The attention of the region aggregation block is unimodal and much more lighter compared to correspondence block.

Table 6: Quantitative comparison on AppearancePointers-27K (Image Region Description with Background). Best result in gold. Higher is better for all metrics. *AppearancePointers* **outperforms** Iterative InsertAnything in **CLIP-I, CLIP-T, MIoU & CLIP-IQA** and performs **competitively on DINO-I even with a setting that favors InsertAnything.**

	Region				Global	
	CLIP-I↑	CLIP-T↑	DINO-I↑	MIoU↑	CLIP-T↑	CLIP-IQA↑
Iterative InsertAnything Song et al. (2025)	94.52	27.46	76.32	40.63	29.74	85.82
***AppearancePointers* (Ours)**	94.52	27.69	75.11	45.44	30.03	95.47

Table 7: **Ablation study on image generation using image-conditioned prompts.** We evaluate the contribution of each component in the *Appearance Pointer* model using the image-based regional description setting. Removing the appearance pointer aggregation significantly harms identity preservation (lower CLIP-I and DINO-I), while slightly improving MIoU due to the absence of region fusion. Omitting the appearance pointer mask reduces spatial disentanglement, yielding weaker region alignment. Disabling position ID resampling degrades both identity and region consistency, especially in multi-region scenes. Finally, removing region contour guidance leads to drop in MIoU. Together, these results highlight the importance of each component and confirm that the full appearance pointer formulation achieves the strongest balance of regional accuracy, spatial adherence, and identity preservation.

	Region				Global	
	CLIP-I↑	CLIP-T↑	DINO-I↑	MIoU↑	CLIP-T↑	CLIP-IQA↑
w/o Region Aggregation section 3.1	89.93	27.35	54.47	41.49	28.45	93.48
w/o Appearance Pointer Mask section B.3	93.23	27.25	68.93	41.09	29.60	95.83
w/o Region Contour Guidance section 3.2	85.57	23.89	41.62	35.81	28.89	94.47
w/o Position ID Resampling Multi-Region Mimic section B.2	93.01	27.2	67.46	40.75	29.47	95.34
***AppearancePointers* (Ours)**	93.29	27.25	69.31	40.97	29.62	95.57

C Multi-Subject Insertion (Editing)

A benefit of our modular region approach is that we can treat the background image and global text prompt as a region condition and seamlessly extend our approach to *an editing task* that allows multi-subject insertion (see Algorithm 1).

We therefore benchmark our method against *InsertAnything* Song et al. (2025), which supports object insertion within a scene image. Since *InsertAnything* can only insert one object at a time, we construct a baseline by iteratively inserting objects. This procedure gives *InsertAnything* a conceptual advantage compared to our method, which inserts all objects simultaneously.

Table 6 and Figure 7 show quantitative and qualitative results comparing our method against *InsertAnything*-based baseline. *AppearancePointers* outperforms iterative *InsertAnything* on CLIP-I, CLIP-T, MIoU, and CLIP-IQA, while remaining competitive on DINO-I, despite handling all regions at once, rather than resorting to an iterative approach. Moreover, the quality of the *InsertAnything*'s generated images deteriorates as the number of insertion operations increases.

D Ablations

Table 7 presents ablations of our *AppearancePointers* method for the image-conditioned prompt setting. We use the image-conditioned prompt setting to evaluate the effectiveness of our method in preserving identity.

w/o Appearance Pointer Aggregation:. We remove the aggregation step, and instead of *Appearance Pointers* ${}^{T}\mathcal{AP}$ and ${}^{I}\mathcal{AP}$, directly use ${}^{T}\mathcal{M}$ and ${}^{I}\mathcal{M}$. For these tokens, we then use the same LoRA fine-tuning approach as for the *Appearance Pointers* tokens. Without the aggregation step to *Appearance Pointers* tokens, the identity of the objects is not well preserved, as indicated by the lower DINO-I (54.47) and CLIP-I (89.93) scores, but not performing Appearance Pointer aggregation enables finer region control with higher MIoU (41.49) as the model does not need to learn how to aggregate regions.

w/o Appearance Pointer Mask:. The masking strategy, described in section B.3, enhances subject identity preservation by ensuring that condition region tokens attend only to tokens within the target

regions. This results in a higher MIoU of 69.31, compared to the baseline without the strategy, denoted in the table as *w/o Appearance Pointer Mask*.

w/o Position ID Resampling Multi-Region Mimic:. The position ID sampling described in section B.2, improves identity for larger number of regions and region adherence without which we obtain a lower DINO-I score of 67.46 and a lower MIoU of 40.75

w/o Region Contour Guidance:. Next, we perform an ablation study by removing the Region Contour Guidance. Table 5 shows that Region Contour Guidance improves all region metrics, increasing MIoU from 35.81 to 40.97.

Performance vs. Number of Regions: table 8 evaluates performance as the number of regions increases from 1 to 9 for image+region to image task. While identity preservation (CLIP-I) remains remarkably stable, regional DINO-I and mIoU show a gradual decline. This is a consequence of increased spatial occlusions and overlapping boundaries in crowded scenes.

Table 8: Precision vs. Number of Regions.

	Metrics / Num Regions	1	2	3	4	5	6	7	8	9
Region	CLIP-T	30.50	29.81	29.5	28.96	28.39	28.19	26.99	26.41	25.13
	CLIP-I ↑	95.07	95.41	95.41	95.51	94.87	95.13	94.84	95.64	94.94
	DINO-I ↑	85.22	85.49	85.56	84.03	82.03	80.45	78.58	76.96	74.80
	MIoU ↑	51.19	51.95	51.88	49.13	48.35	48.05	46.33	44.13	43.63
Global	CLIP-T ↑	34.87	34.96	34.90	34.98	36.27	36.20	35.19	35.87	35.87
	CLIP-IQA ↑	94.62	92.96	93.65	93.97	94.38	95.02	93.07	93.65	93.18

Robustness to Mask Precision: table 9 analyzes model performance across varying mask granularities (Fine vs. BBox vs. Ellipse). Our results demonstrate high stability in identity preservation (CLIP-I) and aesthetic quality (CLIP-IQA), even when using sparse inputs. The decrease in DINO-I and mIoU for coarse masks reflects the model's flexibility in synthesizing novel subject orientations that respect the coarse spatial regions (ellipse & bbox).

Table 9: Coarse Region Control Ablation.

Ablations	Region				Global	
Coarse/Fine Regions	CLIP-T	CLIP-I	DINO-I	MIoU	CLIP-T	CLIP-IQA
Blobs (Ellipse)	28.60	93.72	76.53	44.94	34.93	93.91
Bounding Box (Bbox)	28.79	94.31	80.32	47.86	34.91	93.02
Fine	28.93	95.13	84.03	53.87	35.00	93.51

D.1 Appearance Pointer Pseudocode

Please see Algorithm 1 pseudocode for a detailed pseudocode for implementing attention pointer.

E Qualitative Results

fig. 9, fig. 10, fig. 11, fig. 12, and fig. 13 show visuals of using our method with all conditions together, including material, image, and text conditions. We also perform in-the-wild image editing (fig. 14) and layout-guided generation (fig. 15) on Google images.

Algorithm 1 Appearance Pointer Conditioned Image Generation

Require: ${}^{\mathrm{R}}\mathcal{P}$(Region Masks), ${}^{\mathrm{I}}\mathcal{P}$(Region Images), ${}^{\mathrm{T}}\mathcal{P}$(Region Text Prompts)
Require: ${}^{I}\mathcal{P}_{bg}$(Background Image), timesteps (Generation Timesteps)
Require: R(Number of Regions), ${}^{\mathrm{G}}\mathcal{P}$(Global Prompt)
Ensure: $I :=$ AppearancePointerConditionedGeneration(${}^{\mathrm{R}}\mathcal{P}, {}^{\mathrm{I}}\mathcal{P}, {}^{\mathrm{T}}\mathcal{P}$)

${}^{I}M_s :=$ List(), ${}^{T}M_s :=$ List()
${}^{R}P$.append(${}^{R}\mathcal{P}_{bg}$), ${}^{I}\mathcal{P}$.append(${}^{I}\mathcal{P}_{bg} \odot {}^{R}\mathcal{P}_{bg}$), ${}^{T}\mathcal{P}$.append(${}^{G}\mathcal{P}$) ▷ Add Masked Global Image Prompt and Text Prompt for Editing Application
for $i = 1, 2, \ldots R+1$ **do**
    ${}^{I}M_i, {}^{T}M_i := \mathbf{\Phi}_{RC}([{}^{R}\mathcal{P}_i, {}^{I}\mathcal{P}_i, {}^{T}\mathcal{P}_i])$ ▷ Region-Prompt Linking where Learnable Embedding is Used if Image or Text Prompt is Missing
    ${}^{I}M_s$.append(${}^{I}M_i$), ${}^{T}M_s$.append(${}^{T}M_i$)
end for
${}^{T}\mathcal{M} :=$ Concat(${}^{T}M_s$), ${}^{I}\mathcal{M} :=$ Concat(${}^{I}M_s$)
if AppearancePointerMask **then** ▷ If Appearance-Pointer Mask
    ${}^{T}\mathcal{AP} := \Phi_A^T([{}^{T}A, {}^{T}\mathcal{M} \odot {}^{R}\mathcal{P}])$
    ${}^{I}\mathcal{AP} := \Phi_A^I([{}^{I}A, {}^{I}\mathcal{M} \odot {}^{R}\mathcal{P}])$
else
    ${}^{T}\mathcal{AP} := \Phi_A^T([{}^{T}A, {}^{T}\mathcal{M}])$
    ${}^{I}\mathcal{AP} := \Phi_A^I([{}^{I}A, {}^{I}\mathcal{M}])$
end if
$E :=$ VAE(${}^{R}\mathcal{P}$) ▷ Get Contour Latent
$x_{\text{timesteps}} := \mathcal{N}(0, I)$
for $t =$ timesteps, timesteps $- 1, \ldots 0$ **do**
    $X_t := [x_t; {}^{I}\mathcal{AP}; {}^{I}\mathcal{P}_1; \ldots; {}^{I}\mathcal{P}_n]$ (Image Stream)
    $c := [{}^{G}\mathcal{P}; {}^{T}\mathcal{AP}; {}^{T}\mathcal{P}_1; \ldots; {}^{T}\mathcal{P}_n]$ (Text Stream)
    $x_{t-1} :=$ FLUX(X_t, c, t, E) (Add Contours E to Generative Model)
end for
I := Decode(x_0) ▷ Decode Generated Image
return I

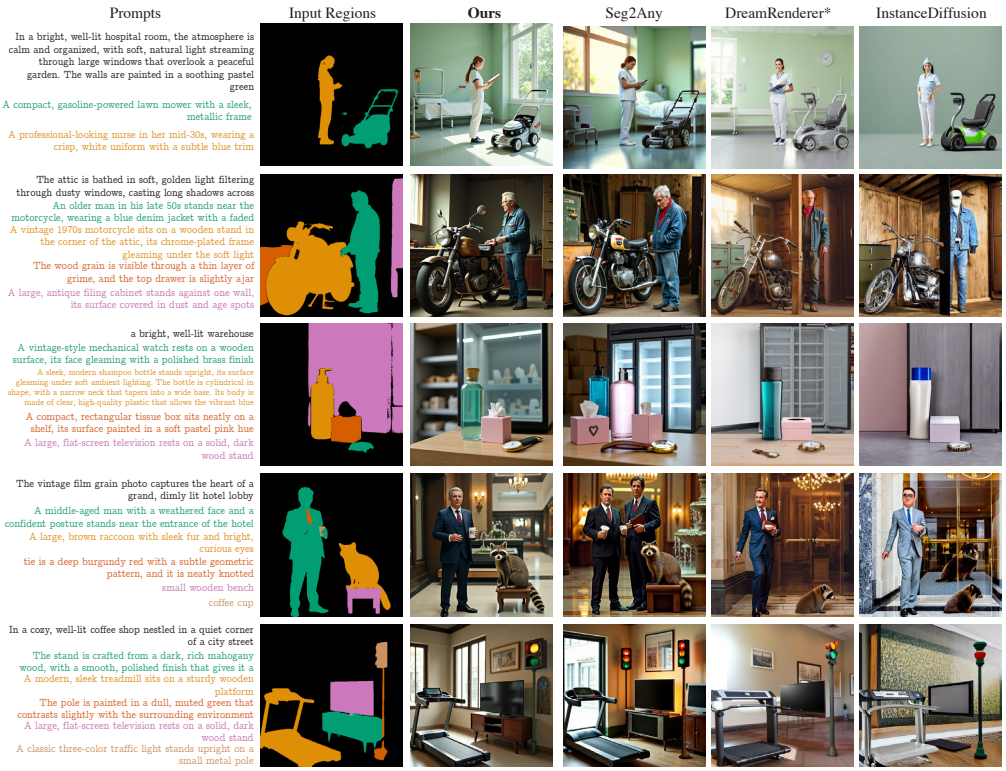

Figure 8: **Qualitative comparison on region-conditioned generation from textual descriptions.** For each example, the left column shows the user-provided regional text prompts (color-coded for clarity), followed by the corresponding input region masks. We compare against Seg2Any, DreamRenderer*, and InstanceDiffusion, all of which support region-level textual guidance. Across diverse scenes—including indoor environments, object–person interactions, retail shelves, and multi-object arrangements—our results show stronger regional fidelity, more accurate object appearance, and better alignment with both global and local textual cues. DreamRenderer* additionally uses depth information, giving it access to geometric cues not available to other baselines. Despite this, our model produces images with higher semantic correctness and more consistent placement across regions.

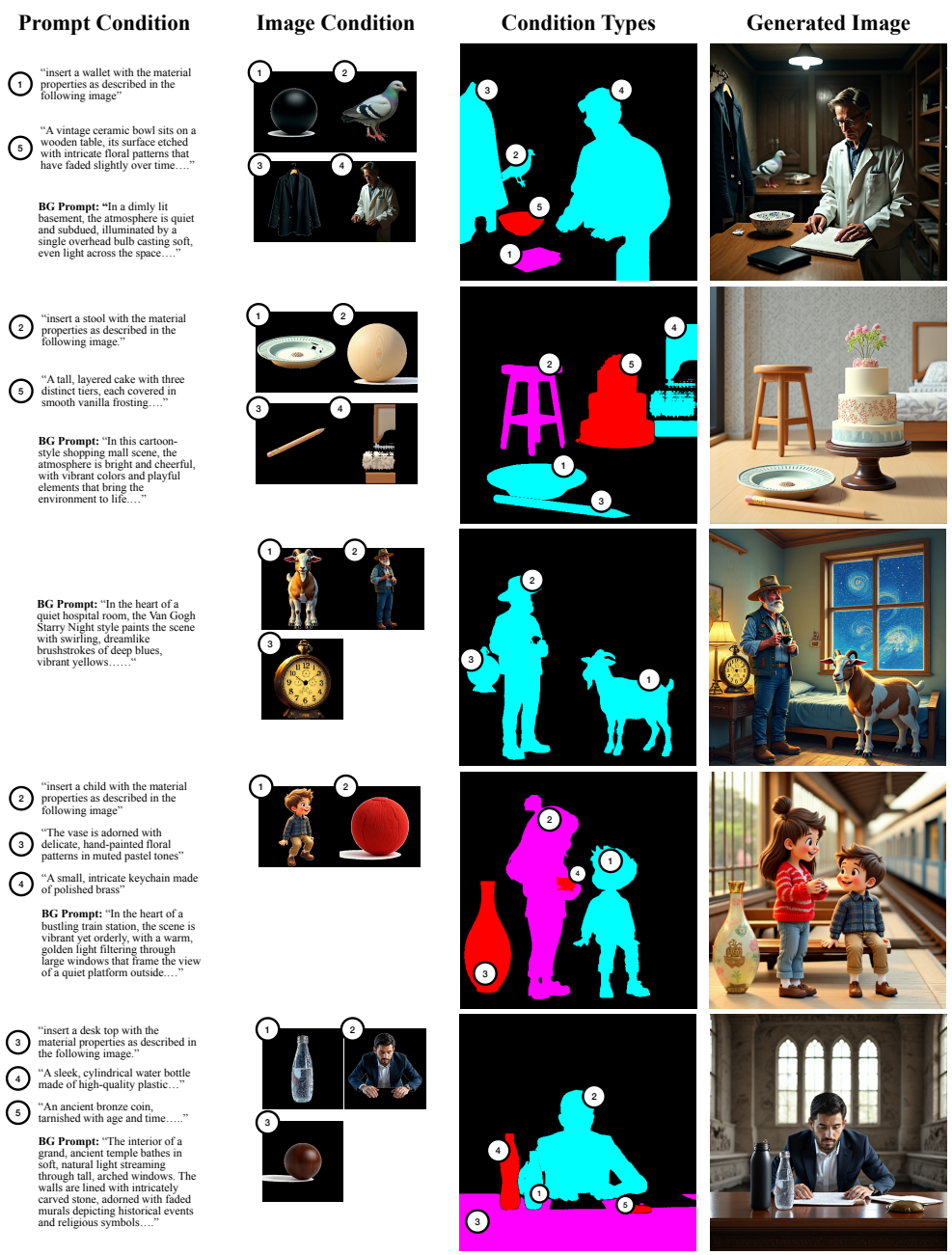

Figure 9: **MultiModal Region Conditioned Generation.** *AppearancePointers* can generate images from heterogeneous condition signals that include materials, images and text prompts. Each region is numbered with the same number as the conditioning signal. The background is generated by the background description. Material conditioned regions are highlighted with bright purple color, image conditioned regions are highlighted with cyan color, and text conditioned regions are highlighted with red color in column 3.

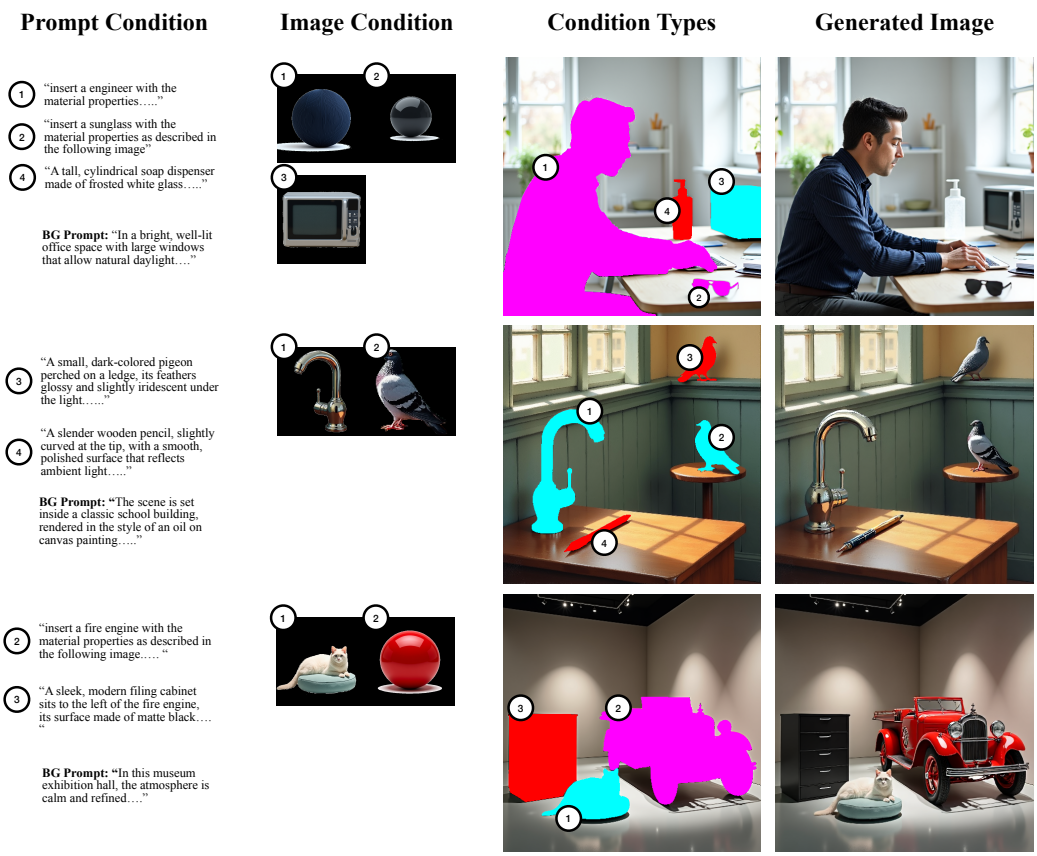

Figure 10: **MultiModal Region Conditioned Insertion.** *AppearancePointers* can insert subjects from heterogeneous condition signals that include materials, images and text prompts. Each region is numbered with the same number as the conditioning signal. The background is unchanged for the editing scenario. Material conditioned regions are highlighted with bright purple color, image conditioned regions are highlighted with cyan color, and text conditioned regions are highlighted with red color in column 3.

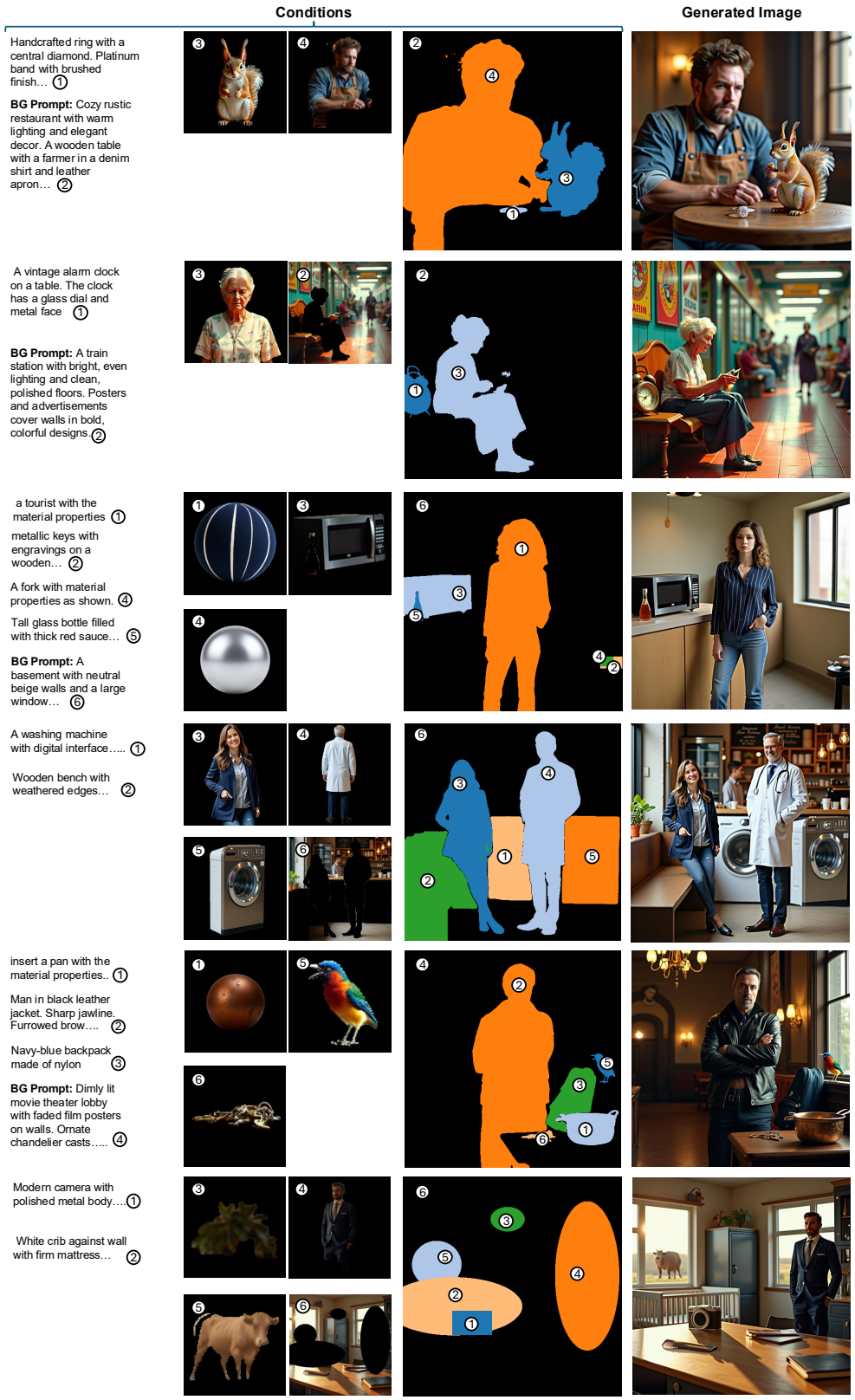

Figure 11: **Image Generation and Editing Gallery** *AppearancePointers* can perform regional edits and generation using heterogeneous conditions in the wild. Each region is accompanied by a number which relates the condition to the region.

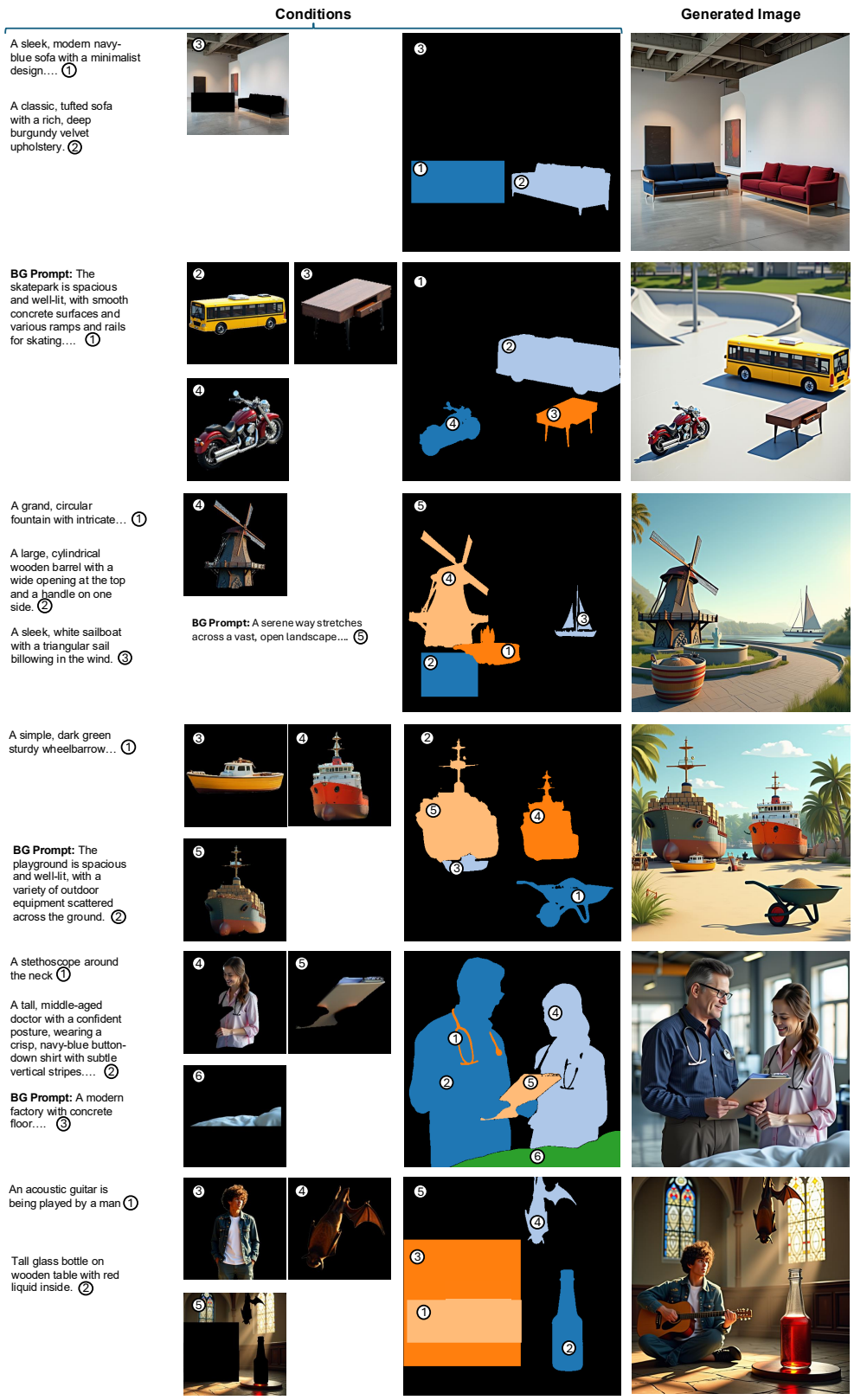

Figure 12: **Image Generation and Editing Gallery** *AppearancePointers* can perform regional edits and generation using heterogeneous conditions in the wild. Each region is accompanied by a number which relates the condition to the region.

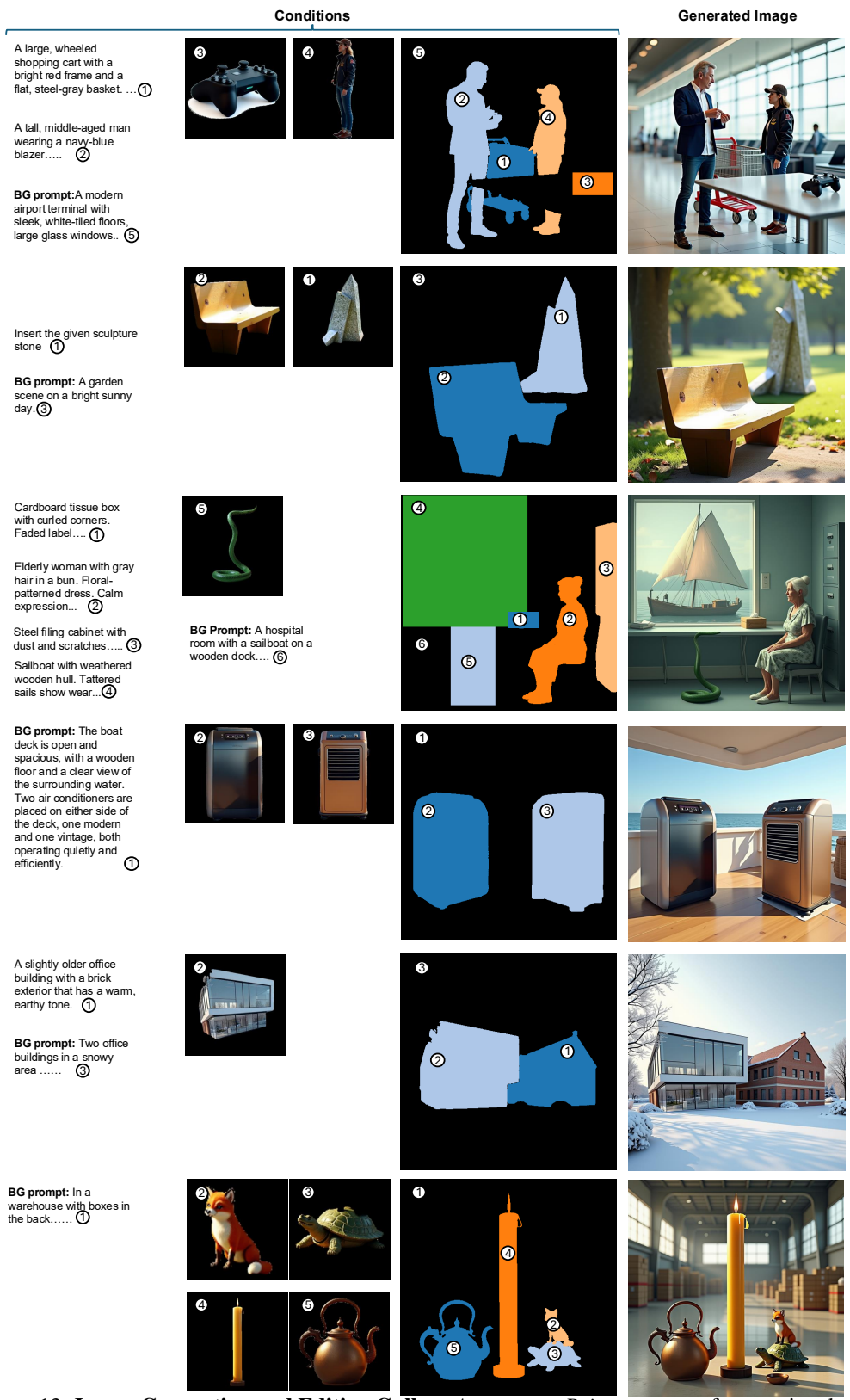

Figure 13: **Image Generation and Editing Gallery** *AppearancePointers* can perform regional edits and generation using heterogeneous conditions in the wild. Each region is accompanied by a number which relates the condition to the region.

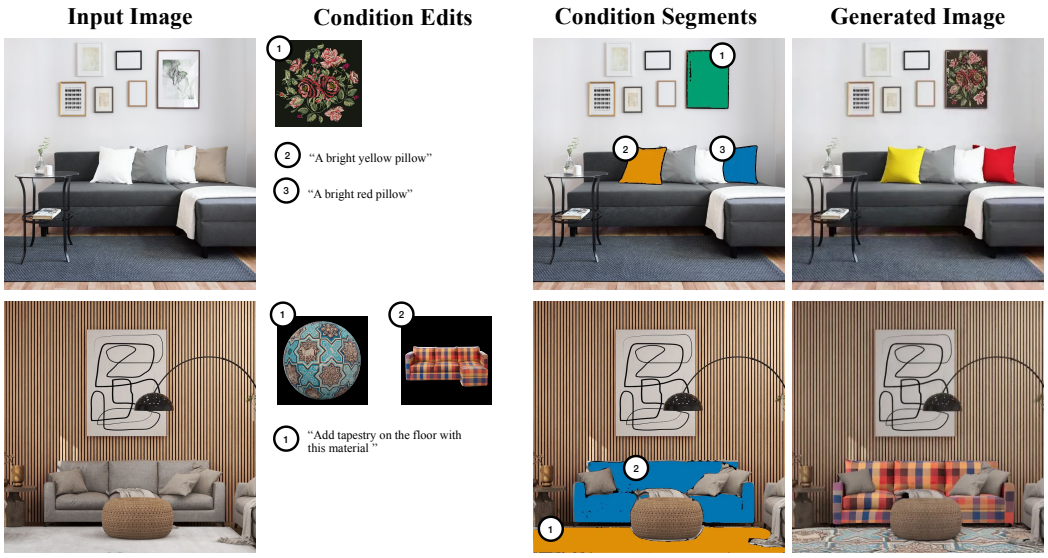

Figure 14: **Image Editing with Images and Region Prompts.** *AppearancePointers* can perform one-shot regional edits on images from the internet.

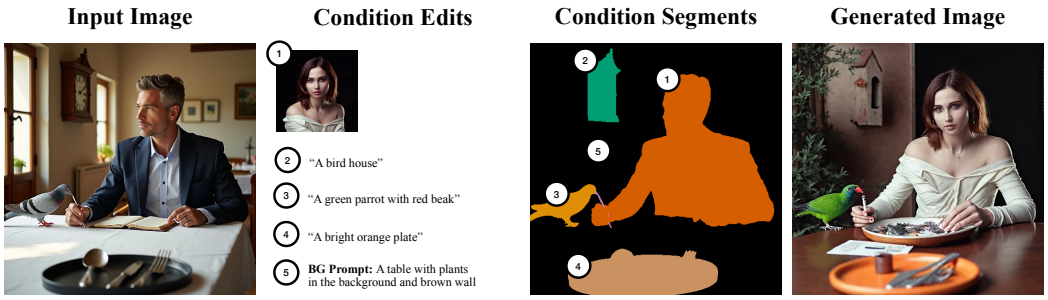

Figure 15: **Layout Guided Generation.** *AppearancePointers* can perform layout guided image generation. In this example, an image segmentation model is used to identify region of an input image. Then, we select a subset of these regions and generate a new image following the same layout. Region 1 is described by a new subject image, while the other regions have text descriptions.

F Evaluation Metrics

Our evaluation benchmark consists of generating 500 images under two conditions: text-only and image-only, both guided by global prompts with an average of 5 regions per image. For the image-only condition, the benchmark additionally includes novel viewpoints of the objects to be inserted into the specified regions. We evaluate using two *global metrics*: (i) **CLIP-IQA** Wang et al. (2023a), which assesses visual quality by comparing the CLIP embedding of the generated image with a 'quality' prompt; (ii) **CLIP-T (Global)**, which measures alignment between the generated image and the global text prompt via the cosine similarity of their CLIP embeddings. We further evaluate *region precision* using four metrics: (i) **DINO-I**, which measures fidelity by computing the cosine similarity between the CLS tokens of DINO features from the generated and ground truth image patches; (ii) **CLIP-I**, analogous to DINO-I but using CLIP image embeddings; (iii) **CLIP-T**, which assesses alignment between the generated image region and its corresponding region prompt via cosine similarity of CLIP embeddings; (iv) **Class-agnostic MIoU** Zhang et al. (2025b), computed by prompting SAM2 on the ground truth regions within the generated image to evaluate shape consistency.

G Baseline Comparison Setting

Following Seg2Any, MS-Diffusion, and InstanceDiffusion, we evaluate baselines using their original optimized weights. Retraining these models on our Appearance Pointers-37K dataset is avoided for the following reasons:

(1) *Metric Neutrality:* Our benchmark utilizes distribution-neutral metrics (mIoU, DINO-I, CLIP-I, CLIP-T, CLIP-IQA) to measure spatial+text adherence, identity preservation and aesthetics. We explicitly exclude distribution-sensitive metrics (FID/KID) to minimize inherent evaluation advantage. We recognize that a residual distribution-shift advantage may remain, and therefore also evaluate on real data (table 4, main paper). (2) *Large-Scale Training:* Prior works such as Seg2Any and MS-Diffusion train at large scale with millions of data samples. Retraining them risks creating weak baselines and is avoided by all prior works as well. Notably, our method **utilizes the same model** for **both image- and text-based regional descriptions** — a significantly more difficult setting that is not supported by prior works. On *SACap-eval* only, for the image-conditioned region comparison against MS-Diffusion we crop each ground-truth object segment to serve as the per-region reference, since real-world data provides no separate references — the only way to supply comparable visual references to both models. On our *AppearancePointers*-37K, reference images are novel views of the objects, so no such cropping is needed. We report quantitative real-world results on SACap-eval in the main paper (table 4) and show qualitative examples below (fig. 16(c)). With light fine-tuning on only 2.3% real data, *AppearancePointers* follows real-world region masks and prompts across diverse scenes and objects (fig. 17).

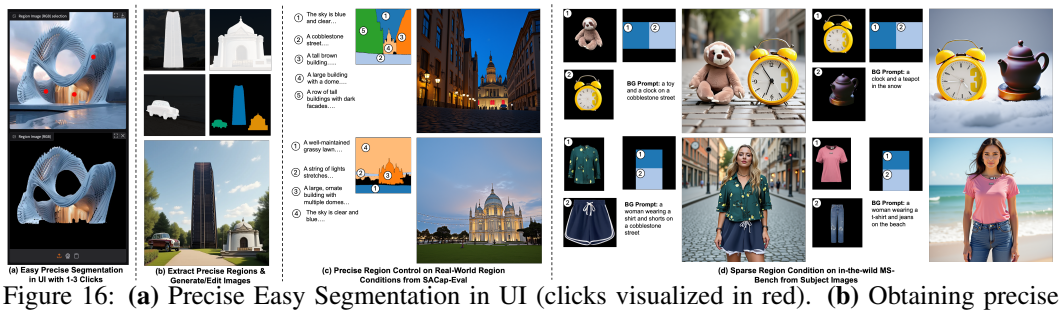

Figure 16: **(a)** Precise Easy Segmentation in UI (clicks visualized in red). **(b)** Obtaining precise regions from other images using a single click and using them as conditions for generating/editing images. **(c)** Real-world regions+text to image generation zero-shot with AppearancePointers on SACap-eval. **(d)** Sparse Bbox subject conditioned image generation from MS-Bench.

Unlike *AppearancePointers*, state-of-the-art text-to-image models lack a native region/mask interface: even when the region masks and descriptions are supplied through in-context prompting, they generate a plausible scene but do not follow the specified spatial layout, as shown against Nano-Banana in fig. 18. To even elicit a layout-conditioned attempt, we provide the mask and task through



Figure 17: **Precise region control on real-world region conditions from SACap-eval.** For each example we show the numbered region prompts (left), their colored region masks (middle), and *AppearancePointers*'s generation (right), after light fine-tuning on only **2.3%** real data. *AppearancePointers* follows the real-world layouts and text while remaining faithful to each region.

in-context prompting — writing a number per segment and asking the model to perform the edits in a multi-step fashion — which required 2–3 edits per image on average, as the model would otherwise render the segment numbers into the output.

H Limitation & Discussion with Future Works

First, our method inherits the FLUX Kontext model's difficulty in preserving intricate human facial identities, a limitation that likely reflects the constraints of the FLUX training distribution. Empirically, we observe that the model preserves identity more reliably when faces occupy larger spatial regions rather than smaller ones. Second, because our dataset verification relies on large-scale VLM-based filtering, residual errors in the verification process may persist and potentially degrade appearance-transfer performance. Finally, conditioning on a large set of regions (about 10) can lead to leaking artefacts or degrade regions, as shown in fig. 19.

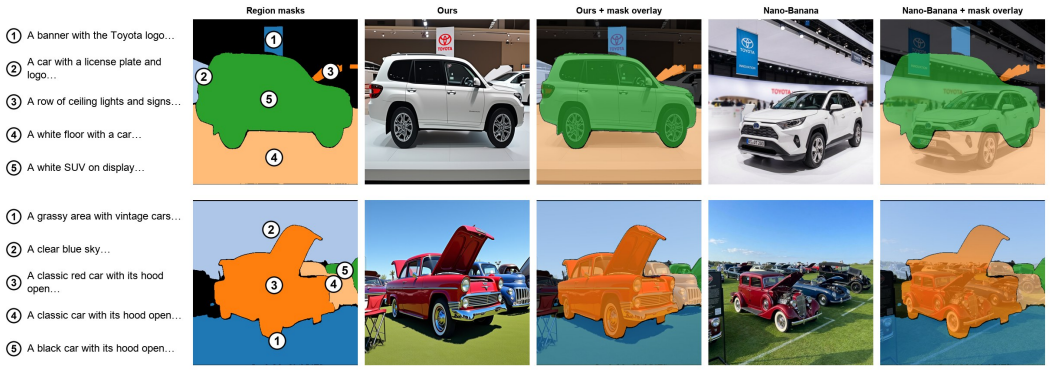

Figure 18: **Text-to-image models do not follow region layouts.** Given the numbered region prompts and masks (left), *AppearancePointers* places each object in its region; overlaying the target masks at 70% alpha shows tight adherence. Even when the same region masks and descriptions are supplied to Nano-Banana through in-context prompting, it produces a plausible image but does not honor the specified spatial layout.

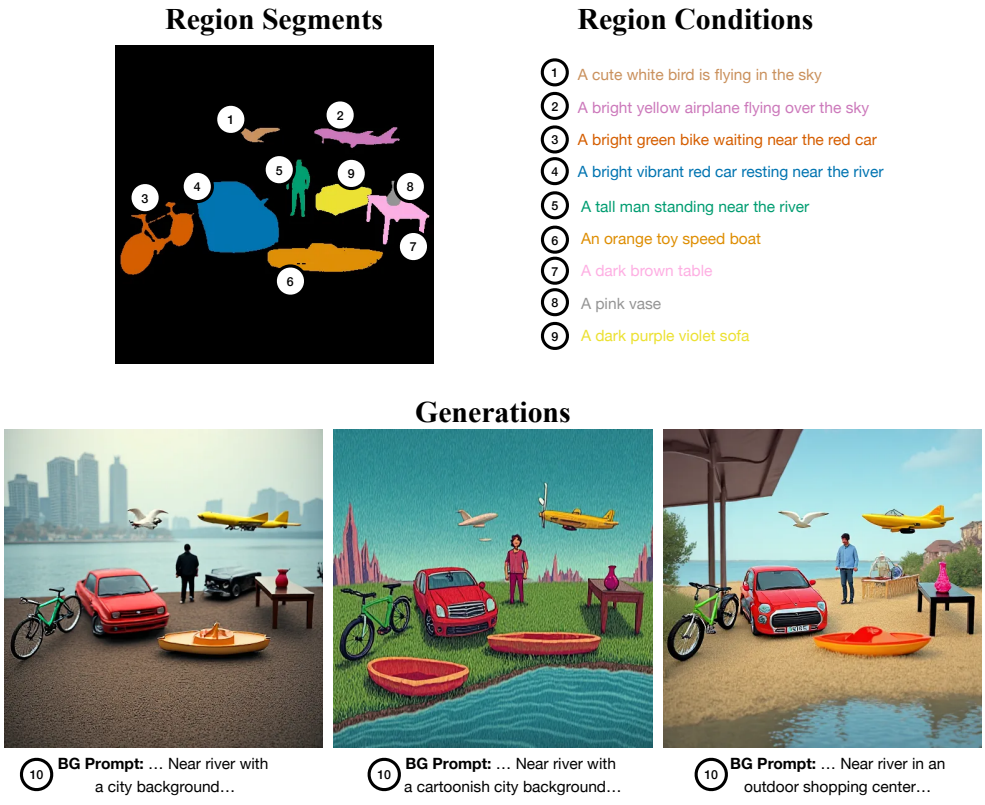

Figure 19: **Failure Cases.** We test scaling the number of regions using text prompt for *AppearancePointers*. Increasing number of regions upto 10 in this case can degrade region adherence and quality. Here, the sofa is not captured accurately.

Our work *AppearancePointers* introduces a method that allows *Pointing* text/image/edit conditions to the desired region and shows competitive or better performance compared to unimodal prior works. A future work for our generic design could be to extend our work to other forms of conditioning modalities, for example, audio, video, 3D, and 4D. Another possible direction is improving the efficiency of the model to make it real-time for editing and scene control applications.

I Appearance Pointer Attention Map Analysis

We provide the attention map visualization in Figure 20 to understand what the pointer tokens are attending to during generation. The yellow queried points on the generated image and segments are queried to obtain the attention map reflecting the regions responsible for conditioning the segment. In each case, our appearance pointer highlights the region of correct region of interest. For example, the pattern on the chair, the right foot of the bench, the lamp, and the crane body each point to the corresponding conditioning signal.

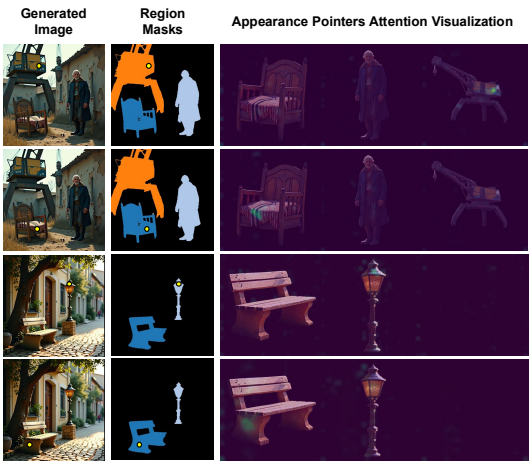

Figure 20: Appearance Pointer *Points* to Correct Regions of Interest displaying its efficacy in multimodal region-controlled generation.

J Additional Dataset Details

J.1 Dataset VLM Check

Edits by FLUX Kontext may not necessarily be accurate and can have inconsistencies that can result in an erroneous signal for our model. Hence, we additionally use InternVL 3 VLM to rate the quality of the edits and prompt it to provide a summary of the reason for the rating (see examples in listing 1 & listing 2).

J.1.1 Novel View Verification

Listing 4 displays an example of prompting the InternVL Chen et al. (2024d) model to score the quality of the novel view generation of the subject of interest. Listing 1 displays the result obtained from the model to rate the overall consistency of the generated novel view.

Listing 1: Example Novel Pose Verification JSON

```
{
   "overall_consistency_score": 0.8,
   "is_geometrically_consistent": true,
   "sharpness_comparison_score": 0,
   "is_novel_pose": false,
   "overall_consistency_reason": "Good Match (Clear resemblance, but minor, subtle
       inconsistencies in material or lighting are noticeable).",
   "geometric_consistency_reason": "N/A",
   "sharpness_comparison_reason": "N/A",
   "pose_novelty_reason": "N/A"
}
```

J.1.2 Material/Texture Verification

Listing 5 displays an example of prompting the InternVL Chen et al. (2024d) to score the quality of the material/texture sphere generated by FLUX-Kontext for the subject of interest. Listing 2 displays the result obtained from the InternVL model to rate the quality of the generated material with its reasoning.

Listing 2: Example Material Verification JSON

```
{
   "perceptual_similarity_score": 0.7,
   "albedo_color_fidelity_score": 0.8,
   "roughness_gloss_fidelity_score": 0.6,
   "texture_detail_fidelity_score": 0.8,
   "albedo_color_reason": "Sphere's color is slightly lighter than the object.",
   "roughness_gloss_reason": "Sphere appears smoother than the object's material.",
   "texture_detail_reason": "Sphere's pattern is similar but lacks ....",
```

```
   "perceptual_consistency_reason": "Good match but some differences in texture depth and
       glossiness."
}
```

J.2 PROMPTING

Here, we detail the additional details that we used to prompt LLM. Listing 3 displays the JSON prompt provided to Qwen 3 Yang et al. (2025a) to generate object descriptions, material descriptions, Grounding DINO captions, and more. In our work, we found succinct Grounding DINO captions necessary to improve ground score and succinct material descriptions to prompt FLUX Kontext Labs et al. (2025) to improve material/texture sphere generation.

Listing 3: Example for Generating Prompts and Object Descriptions for our Dataset

```
{
   "task": "expert scene captioning and material extraction for 3D generation",
   "style": "{style}",
   "environment": "{environments}",
   "required_object": "{prompt_obj_string}",
   "people_details": "{people_detail_description}",
   "constraints": {
      "output_format": "JSON object",
      "total_word_count_min": 1000,
      "total_word_count_max": 3000,
      "scene_caption_word_count": 300,
      "image_quality": "avoid dark, low-resolution, or overly complex scenes",
      "visual_distinctiveness_rule": "ALL objects generated in the scene must be visually
          differentiable. Ensure colors, primary textures, and unique features DO NOT overlap
           between objects. For example, if Object A is 'smooth, dark mahogany,' Object B
          must not be 'smooth, dark walnut.'",
      "material_specificity_rule": "Material descriptions must be vivid, focusing on sensory
          qualities: the degree of shine, the feel of the texture, and how light interacts
          with the surface. Make sure you mention description of the dominant color and any
          texture elements. Avoid technical 3D terms."
   },
   "instructions": [{
      "step": 1,
      "name": "detailed_object_captions",
      "goal": "Generate a detailed, purely visual description (max 100 words per object) for
          each object listed in 'required_objects'. These descriptions must be entirely
          isolated: contain NO environmental context, NO relation to neighboring objects, and
           NO scene placement. Focus only on intrinsic visual attributes.",
      "subtasks": ["For the 'caption' field, detail color, shape, and intricate surface
          patterns.",
               "For the 'dino_caption' field, generate a succinct, precise phrase (under 10
                   words) focusing on the object's function or unique geometry to maximize
                   detection accuracy, avoiding reliance on simple color.",
               "**If the object is a person, the 'material_caption' must describe the
                   material and texture of their most prominent article of clothing (e.g.,
                   shirt, jacket, or pants), detailing its weave, reflectivity, and drape.**.
                    For all other objects, provide a rich, sensory and perceptual description
                    of the object's material properties, describing texture, reflectivity,
                   and light interaction in the material caption with the **single** most
                   prominent article of the material along with its color (under 10 words).
                   DO NOT describe multiple materials or colors."
      ]},
      {
      "step": 2,
      "name": "scene_caption",
      "goal": "Generate a final, vivid scene caption by seamlessly combining all objects from
          Step 1. Focus on **scene-level details** (placement, lighting, composition, object
          interaction) within the specified **environment** and **style**. Ensure all objects
           are in sharp focus and not part of a blurred background. The caption must be upto
          **300 words** long."
      }
   ],
   "output_structure": {
      "detailed_object_captions": {
      "object_1":
         {
            "name": "<object_1_name>",
            "caption": "<detailed object 1 visual caption>",
            "dino_caption": "<succinct dino caption for object 1 for better detection>",
            "material_caption": "<detailed visual and textural description of the object's
                material>"
         },
```

```
      "object_2":
         {
            "name": "<object_2_name>",
            "caption": "<detailed object 2 visual caption>",
            "dino_caption": "<succinct dino caption for object 2 for better detection>",
            "material_caption": "<detailed visual and textural description of the object's
                 material>"

         },
      },
      "scene_caption": "<scene caption, upto 300 words long>",
      "scene_background_caption": "<scene background caption, upto 300 words long>"
   }
}
```

Listing 4: Example for Rating the Quality of Novel View Pose

```
{
   "task": "Expert simplified 3D view consistency and critical artifact analysis",
   "goal": "Given two images of the same object from different viewpoints, provide a single
        overall consistency score and three critical binary/categorical checks (Geometric,
        Sharpness, Pose). The ONLY output must be a single, valid JSON object that adheres
        strictly to the defined output_structure, with absolutely no preamble, explanation, or
         concluding text.",
   "prompt": "Analyze the two object views. Assume both images are renders of the *same 3D
        model* and *same material*. **CRITICAL RULE: IGNORE ALL PIXELATION AND BLOCKING
        ARTIFACTS** when determining scores. On a scale from 0.0 (inconsistent) to 1.0 (
        perfectly consistent), score the **overall visual consistency**, encompassing
        perceptual match, material realism, and texture detail. **GEOMETRIC CHECK:** Set **
        is_geometrically_consistent** to **true** if the 3D shape/scale is plausible and
        consistent, and **false** if there are significant errors. Provide an **integer score
        ** for the sharpness comparison based on the rule: **0** if SAME SHARPNESS; **-1** if
        View 1 is MORE BLURRY; and **+1** if View 2 is MORE BLURRY. Additionally, set **
        is_novel_pose** to **true** if View 1 and View 2 have a DIFFERENT POSE or ORIENTATION,
         and **false** otherwise. Provide a brief, one-sentence justification for **every
        output**. **MAKE SURE YOU STICK TO THE RUBRIK AND FOLLOW IT EXACTLY**. **IGNORE THE
        WHITE BACKGROUND IN BOTH IMAGES. JUST FOCUS ON THE OBJECT ITSELF**.",
   "scoring_rubric": {
      "overall_consistency_score": {
         "0.9-1.0": "Excellent Match (Nearly indistinguishable; excellent consistency across
              all lighting, texture, and perceptual aspects, ignoring pixelation).",
         "0.7-0.89": "Good Match (Clear resemblance, but minor, subtle inconsistencies in
              material or lighting are noticeable).",
         "0.4-0.69": "Fair Match (Obvious inconsistencies in one or more areas, such as color
              shift or smeared textures).",
         "0.0-0.39": "Poor Match (Gross inconsistencies; views look like different objects or
              materials)."
      },
      "is_geometrically_consistent": {
         "true": "3D shape, scale, and placement are plausible and consistent (minor artifacts
               acceptable).",
         "false": "Significant geometric errors are present (distortion, object shifting, hole
              -filling errors, or wrong scale)."
      },
      "sharpness_comparison_score": {
         "+1": "View 2 is noticeably softer, has less fine detail, or appears more out-of-
              focus than View 1 (View 2 is the blurrier image).",
         "0": "The perceptual sharpness and fine detail levels are visually identical between
              View 1 and View 2.",
         "-1": "View 1 is noticeably softer, has less fine detail, or appears more out-of-
              focus than View 2 (View 1 is the blurrier image)."
      },
      "is_novel_pose": {
         "true": "**NOVEL POSE**: View 2 is rotated/transformed and is different from view 1 (
              difference is easily visible).",
         "false": "**NO NOVELTY IN POSE**: View 2 is the same orientation as view 1 (
              difference is minimal or invisible)."
      }
   },
   "output_structure": {
      "overall_consistency_score": "float (0.0 to 1.0)",
      "is_geometrically_consistent": "bool (one of: true, or false)",
      "sharpness_comparison_score": "integer (one of: -1, 0, or 1)",
      "is_novel_pose": "bool (one of: true, or false)",
      "overall_consistency_reason": "string (e.g., 'Low score due to clear color shift and
           texture smearing, NOT pixelation' or 'N/A')",
      "geometric_consistency_reason": "string (e.g., 'Significant geometric distortion is
           visible in View 2' or 'N/A')",
      "sharpness_comparison_reason": "string (e.g., 'Score is +1 because fine edges are lost
           in View 2' or 'N/A')",
```

```
    "pose_novelty_reason": "string (e.g., 'Pose is novel because it is rotated 30 degrees
        around up-down axis' or 'N/A')"
  }
}
```

Listing 5: Example for Rating the Material/Texture Quality of the Generated Edit

```
{
  "task": "expert graphics 3D shape and material understanding",
  "goal": "numerically score the fidelity of three distinct PBR attributes between the object
      and sphere images.",
  "prompt": "Analyze the two images (segmented object and sphere). On a scale from 0.0 to
      1.0, score the degree to which the material properties and texture on the sphere
      accurately represent those of the object. Score the **Albedo/Color**, **Roughness/
      Gloss**, **Perceptual Match** and **Texture/Pattern Detail** independently. Output the
       three scores as floating-point numbers. **MAKE SURE YOU STICK TO THE RUBRIK AND
      FOLLOW IT EXACTLY**. **IGNORE THE WHITE BACKGROUND IN BOTH IMAGES. JUST FOCUS ON THE
      OBJECT AND SPHERE ITSELF**.",
  "scoring_rubric": {
    "perceptual_similarity": {
      "0.9-1.0": "Excellent Match (Visually indistinguishable; the sphere perfectly
          captures the 'feel' and realism of the object's material).",
      "0.7-0.89": "Good Match (The sphere clearly represents the intended material, but a
          layman could spot minor flaws like poor lighting interaction or blurring).",
      "0.4-0.69": "Fair Match (The material looks synthetic, plastic, or simply 'off'
          compared to the real object, severely impacting believability).",
      "0.0-0.39": "Poor Match (The material is unrecognizable or looks like a gross
          simplification; fails the basic realism test)."
    },
    "albedo_color": {
      "0.9-1.0": "Excellent Match (Nearly identical color, saturation, and lightness).",
      "0.7-0.89": "Good Match (Correct hue, but lightness or saturation is clearly wrong/
          off).",
      "0.4-0.69": "Fair Match (Significant color shift; related but incorrect hue).",
      "0.0-0.39": "Poor Match (Fundamentally different color)."
    },
    "roughness_gloss": {
      "0.9-1.0": "Excellent Match (Highlights and surface diffusion perfectly match
          observed shininess/dullness).",
      "0.7-0.89": "Good Match (Generally correct, but highlights are slightly too sharp or
          too broad).",
      "0.4-0.69": "Fair Match (Significant misrepresentation of surface type, e.g., shiny
          plastic rendered as dull cloth).",
      "0.0-0.39": "Poor Match (Completely inverted property)."
    },
    "pattern_detail": {
      "0.9-1.0": "Excellent Match (Texture pattern, scale, and orientation are identical)."
          ,
      "0.7-0.89": "Good Match (Correct pattern used, but slightly tiled, scaled, or rotated
           incorrectly).",
      "0.4-0.69": "Fair Match (Right *type* of texture, but the *specific pattern* is
          different or map resolution is poor).",
      "0.0-0.39": "Poor Match (Completely different pattern applied)."
    }
  },
  "output_structure": {
    "perceptual_similarity_score": "float (0.0 to 1.0)",
    "albedo_color_fidelity_score": "float (0.0 to 1.0)",
    "roughness_gloss_fidelity_score": "float (0.0 to 1.0)",
    "texture_detail_fidelity_score": "float (0.0 to 1.0)",
    "albedo_color_reason": "string (e.g., 'Sphere's color is too light' or 'N/A')",
    "roughness_gloss_reason": "string (e.g., 'Sphere is too shiny' or 'N/A')",
    "texture_detail_reason": "string (e.g., 'Sphere's pattern is different than object as it
         contains....' or 'N/A')",
    "perceptual_consistency_reason": "string (e.g., 'Low perceptual similarity because.....
        ' or 'N/A')"

  }
}
```