PLANNING AND RENDERING IN CONCERT: DEEPFUSION OF AUTOREGRESSIVE LAYOUTS AND DIFFUSION FOR VISUAL TEXT GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Generating text-rich images from prompts requires both textual fidelity and the coherent integration of text into the surrounding image. An explicit layout can provide structured guidance about what text should appear and where, but a well-formed plan alone does not guarantee that the renderer will realize it faithfully. Existing layout-based AR–diffusion systems typically optimize planning and rendering separately, preventing the planner's representations from being adapted jointly with image synthesis. We introduce DuetGen, an autonomous visual text generator built on DeepFusion, which jointly learns autoregressive planning and continuous diffusion rendering. DeepFusion conditions a diffusion transformer on the planner's prompt and bbox–content hidden states, allowing rendering supervision to shape the representations connecting textual plans with visual outputs. Its joint objective combines autoregressive plan supervision, text-region-weighted diffusion learning, and auxiliary coordinate supervision to maintain structured planning, emphasize text-bearing regions, and improve the spatial precision of planner representations. During inference, Phase-Aware Attention Modulation strengthens the correspondence between image regions and their matched coordinate and content states, facilitating region-specific execution of the generated plan. With a 2B planner and a 4B single-stream DiT, DuetGen achieves 0.8293 word accuracy on CVTG-2K and 0.938 accuracy on LongText-Bench, closely matching the substantially larger Qwen-Image on both benchmarks. These results demonstrate the value of jointly learned planning representations and region-specific rendering for autonomous visual text generation.

1 INTRODUCTION

Text is ubiquitous in natural scenes and designed media, from signs and posters to book covers and digital interfaces. Generating text-rich images requires a model to determine what text belongs in the scene, organize multiple regions in space, and faithfully render each string at its intended location (Lin et al., 2024; Peng et al., 2025; Wu et al., 2025a). Visual text generation therefore couples content-and-layout planning with rendering in a continuous visual space.

Layout-conditioned diffusion models can synthesize high-quality text-rich images from explicit boxes and strings (Du et al., 2025; Chen et al., 2024c). However, these layouts are typically supplied externally rather than inferred from the prompt. Autonomous generation must instead infer its own layout and use it effectively to guide synthesis.

Two paradigms have been explored toward this goal, as summarized in Figure 1. Separated AR–diffusion systems pass an autoregressively generated text–layout specification to a separately optimized diffusion renderer (Chen et al., 2024a; Zhang et al., 2025a). This two-stage design retains continuous diffusion rendering but lacks deep integration between planning and rendering. Fully autoregressive systems provide tighter integration by modeling layouts and images in a shared sequence (He et al., 2025; Lu et al., 2025; Mao et al., 2026). However, their discrete visual tokens create a representation bottleneck for fine-grained glyph and image synthesis (Yu et al., 2023; Ma et al., 2025a).

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

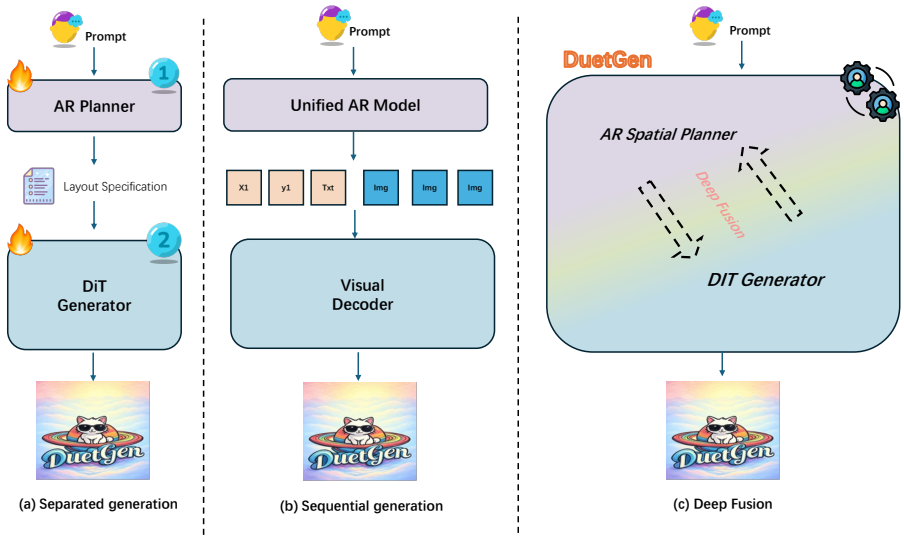


Figure 1: **Paradigms for autonomous layout-aware visual text generation.** (a) Separated AR–diffusion systems separately optimize planning and continuous rendering in two stages. (b) Fully autoregressive systems jointly model layout and image tokens, but rely on discrete visual representations. (c) DeepFusion jointly optimizes AR planning and continuous diffusion rendering through shared planner states and region-specific execution.

These limitations motivate jointly optimizing structured AR planning and continuous diffusion rendering. The planner representations need to preserve structured content and geometry while serving as effective conditions for synthesis; the DiT must then faithfully execute their region-specific information.

We introduce DuetGen, an autonomous visual text generator that realizes this principle through **DeepFusion**. An AR planner produces a bbox–content plan, and its prompt and plan hidden states condition a DiT through a lightweight mapper. A joint planning–rendering objective combines autoregressive plan supervision, text-region-weighted diffusion learning, and auxiliary coordinate supervision. Together, these objectives preserve structured planning, improve the spatial precision of plan representations, and focus rendering learning on text-bearing regions. During sampling, Phase-Aware Attention Modulation (PAM) selectively strengthens attention from patches inside each planned region to its matched bbox and content states. DeepFusion thereby integrates planning and continuous rendering through jointly learned representations, coordinated objectives, and region-specific execution.

Our contributions are summarized as follows:

- We introduce DuetGen, which deeply integrates autoregressive layout planning and continuous diffusion rendering through DeepFusion.
- We jointly learn prompt-and-plan representations through autoregressive plan supervision, text-region-weighted diffusion learning, and auxiliary coordinate supervision, and strengthen region-specific plan execution with PAM.
- Extensive analyses and experiments validate this integration across planner scales and diffusion backbones, achieving competitive multi-region and long-text generation.

2 Related Work

2.1 Visual Text Rendering

Visual text rendering, which aims to seamlessly integrate legible and controllable text into generated images, has attracted increasing attention. Foundational work focused on explicit character-level conditioning, either by incorporating rendered glyphs as priors (Ji et al., 2023; Zhao & Lian,

2024; Ma et al., 2023; 2025b; Yang et al., 2023) or by developing specialized character-aware encoders (Liu et al., 2024a;b). To support multilingual rendering, approaches have leveraged OCR models for stroke features (Tuo et al., 2023; 2024) or employed visual glyph replication (Wang et al., 2025c). Concurrently, methods emerged to control spatial arrangement through attention map manipulation (Xie et al., 2023) and full layout-to-image synthesis (Zheng et al., 2023). More recent advances have pursued higher typographic fidelity and novel generation paradigms. These include efforts to enhance word-level fidelity, style, and alignment (Wang et al., 2025d; Shi et al., 2025b;a), as well as the exploration of layout-agnostic (Zhangli et al., 2024) and training-free, outline-guided generation (Zhang et al., 2024). The problem's scope has also expanded to multi-instance scenarios using agent-based systems (Li et al., 2025b; Zhao et al., 2024) and fine-grained attribute control (Zhou et al., 2025; 2024b). This has led to a surge in domain-specific applications, including specialized models for multilingual text (Jiang et al., 2025; Xie et al., 2025b; Li et al., 2024), posters (Hu et al., 2025; Gao et al., 2025a; Chen et al., 2025e), long-form content (Wang et al., 2025a), ancient scripts (Li et al., 2025a), and broader graphic design automation (Liang et al., 2024; Wang et al., 2025e; Chen et al., 2025a).

These prior works primarily focus on the synthesis stage, assuming layout specifications are provided externally. DuetGen, in contrast, operates end-to-end: it autonomously plans a structured blueprint from a prompt and then renders it, using joint training to better align planning with synthesis.

2.2 Unified Multimodal Modeling

Unified multimodal modeling has evolved towards single architectures handling both understanding and generation by combining autoregressive models for text with diffusion models for images. Transfusion (Zhou et al., 2024a) pioneered this by training a single Transformer on discrete text tokens with causal attention and continuous image patches with bidirectional attention, applying next-token prediction and diffusion objectives simultaneously. Show-o (Xie et al., 2024) builds upon pre-trained LLMs and employs discrete denoising diffusion for image tokens, processing text autoregressively and images via full attention. BAGEL (Deng et al., 2025) employs a decoder-only architecture with dual Transformer experts for understanding and generation, using separate encoders for semantic content and pixel-level information, while BLIP3-o (Chen et al., 2025b) uses a diffusion transformer to generate CLIP image features with sequential pretraining. TransDiff (Zhen et al., 2025) adopts joint end-to-end training with an AR Transformer encoding inputs into semantic features that guide a diffusion decoder.

DuetGen specializes AR–diffusion modeling for visual text generation through joint layout planning and continuous rendering, with objectives and inference-time attention modulation designed for region-specific text synthesis.

3 Method

DuetGen couples a Qwen3.5-2B autoregressive planner (Qwen Team, 2026) with a 4B single-stream DiT through DeepFusion. We pretrain the DiT from random initialization for general image generation before joint planning–rendering training (Appendix A). Given a prompt p, the planner emits a bbox–content plan specifying the position and content of each text region. A lightweight mapper projects the planner's final-layer prompt and plan states into condition tokens for the DiT. The DiT then denoises image latents while attending jointly to these condition tokens. Figure 2 summarizes training and inference.

3.1 Bbox–Content Planning

The plan is $\mathcal{S} = \{(b_i, c_i)\}_{i=1}^{K}$, where $b_i = (x_i^1, y_i^1, x_i^2, y_i^2)$ is a bounding box and c_i is the text to be rendered inside it. We serialize each region as

$$s_i = \big[\langle\text{region}\rangle,\ x_i^1, y_i^1, x_i^2, y_i^2,\ \langle\text{text}\rangle,\ c_i,\ \langle/\text{region}\rangle\big]. \tag{1}$$

We refer to s_i as a *bbox–content sequence*: the four coordinate fields form the coordinate span, and the tokens following $\langle\text{text}\rangle$ form the content span.

Let $s = (s_1, \ldots, s_K)$ denote the complete serialized plan. During training, the planner is teacher-forced on the ground-truth plan; at inference, it generates the plan autoregressively. Its LM head

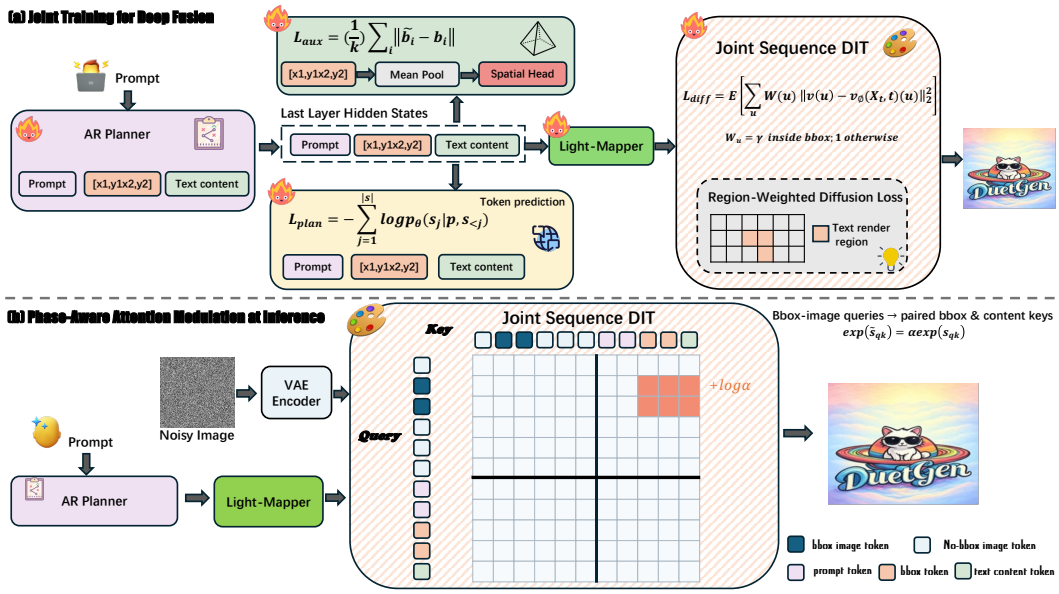


Figure 2: **Overview of DuetGen.** (a) DeepFusion jointly optimizes planning and rendering through prompt-and-plan states, autoregressive plan supervision, text-region-weighted diffusion learning, and auxiliary coordinate supervision. (b) During inference, Phase-Aware Attention Modulation strengthens attention from in-region patches to their matched bbox and content states in middle DiT layers and denoising steps.

maps each hidden state to the next-token distribution p_θ, yielding the autoregressive plan loss

$$\mathcal{L}_{\text{plan}} = -\sum_{j=1}^{|s|} \log p_\theta(s_j \mid p, s_{<j}). \tag{2}$$

3.2 Joint Training

We concatenate the prompt and bbox–content sequence and extract the planner's final-layer hidden states $\mathbf{H}_p$ and $\mathbf{H}_s$ at their respective token positions. A lightweight mapper m_η converts these prompt-and-plan states into DiT condition tokens $\mathbf{C}$:

$$\mathbf{C} = m_\eta([\mathbf{H}_p; \mathbf{H}_s]), \qquad \mathbf{X}_t = [\mathbf{Z}_t; \mathbf{C}]. \tag{3}$$

$\mathbf{Z}_t$ denotes the patch tokens of $\mathbf{z}_t$. The DiT processes $\mathbf{X}_t$ as a single sequence.

For flow-matching training, we sample $\mathbf{z}_t = (1 - \sigma_t)\mathbf{z}_0 + \sigma_t \epsilon$ and predict the velocity $v = \epsilon - \mathbf{z}_0$. We apply a text-region-weighted diffusion loss:

$$\mathcal{L}_{\text{diff}} = \mathbb{E}\left[\sum_{\mathbf{u}} W(\mathbf{u}) \, \|v(\mathbf{u}) - v_\phi(\mathbf{X}_t, t)(\mathbf{u})\|_2^2\right]. \tag{4}$$

Here, $\mathbf{u}$ indexes a spatial location on the latent grid, and $W(\mathbf{u})$ equals γ on the union of rasterized reference boxes and 1 elsewhere. The velocity prediction is evaluated on this grid after unpatchifying the DiT output. Setting $\gamma = 1$ gives the standard flow-matching loss; Appendix C specifies the rasterization.

For auxiliary coordinate supervision, we mean-pool the states of each coordinate span and regress its normalized box:

$$\tilde{b}_i = \sigma(g_\psi(\text{Mean}(\mathbf{H}_{b_i}))), \qquad \mathcal{L}_{\text{aux}} = \frac{1}{K}\sum_{i=1}^{K} \|\tilde{b}_i - b_i^{\text{norm}}\|_1, \tag{5}$$

where $\mathbf{H}_{b_i} \subset \mathbf{H}_s$ denotes the coordinate-span states of region i, and $b_i^{\text{norm}} \in [0,1]^4$ is its reference box normalized by image width and height. The auxiliary head g_ψ is used only during training. The prompt-and-plan states, rather than the box predictions from this head, condition the DiT.

The joint planning–rendering objective is

$$\mathcal{L} = \mathcal{L}_{\text{plan}} + \lambda_{\text{diff}}\mathcal{L}_{\text{diff}} + \lambda_{\text{aux}}\mathcal{L}_{\text{aux}}. \tag{6}$$

We jointly optimize the planner, mapper, DiT, and auxiliary head, while keeping the VAE frozen.

3.3 Phase-Aware Attention Modulation

At inference, Phase-Aware Attention Modulation (PAM) strengthens only the attention from image patches inside each planned region to the bbox and text-content tokens of the same region. The modulation is restricted to the middle DiT layers $\mathcal{M}$ and a denoising window $\mathcal{T}_{\text{mid}}$:

$$\begin{aligned} \tilde{s}_{qk}^{(\ell,t)} &= \frac{\langle Q_q^{(\ell,t)}, K_k^{(\ell,t)} \rangle}{\sqrt{d}} + \Delta_{qk}^{(\ell,t)}, \\ \Delta_{qk}^{(\ell,t)} &= \begin{cases} \log \alpha, & \ell \in \mathcal{M},\ t \in \mathcal{T}_{\text{mid}},\ q \in \mathcal{P}_i^{\text{box}}, k \in \mathcal{K}_i^{\text{box}} \cup \mathcal{K}_i^{\text{text}} \text{ for some } i, \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \tag{7}$$

$\mathcal{P}_i^{\text{box}}$ denotes the indices of DiT image tokens whose patch centers lie inside b_i; for each $q \in \mathcal{P}_i^{\text{box}}$, $Q_q^{(\ell,t)}$ is the corresponding query vector. $\mathcal{K}_i^{\text{box}}$ and $\mathcal{K}_i^{\text{text}}$ index the condition-token positions projected from the coordinate and content spans of region i. For $\alpha > 1$, adding $\log \alpha$ multiplies the unnormalized attention weights of exactly these query–key pairs by α. All other logits are unchanged. Softmax renormalizes each affected query row, so normalized attention to other keys in that row can decrease. PAM thus increases the relative attention assigned to region-matched bbox and content tokens.

4 Experiments

4.1 Experimental Setup

Datasets. We pretrain our randomly initialized 4B single-stream DiT on ImageNet-1K and then on approximately 15M generic image–caption pairs at 512×512 resolution (Appendix A). For joint training, we combine MARIO-10M (Chen et al., 2023), CreateDesign (Zhang et al., 2025b), Lex-10K (Zhao et al., 2025), PrismLayers (Chen et al., 2025d), and LLaVAR-2 (Zhou et al., 2024d); filtering and OCR-based processing yield 6,266,861 text–image pairs with transcriptions and bounding boxes. We evaluate on CVTG-2K (Du et al., 2025), LongText-Bench (Geng et al., 2025), and TextAtlasEval (Wang et al., 2025b).

Implementation details. Joint training starts from Qwen3.5-2B (Qwen Team, 2026) for the planner and our generic-pretrained 4B single-stream DiT for the renderer. We jointly train the planner, mapper, DiT, and auxiliary head for one epoch on 16 NVIDIA A800 GPUs using bfloat16 precision, gradient checkpointing, and DeepSpeed ZeRO-2 (Rajbhandari et al., 2020), with a global batch size of 64; the VAE remains frozen. We use AdamW (Loshchilov & Hutter, 2017): the planner uses a learning rate of 1×10^{-5}, while the DiT, mapper, and auxiliary head use 5×10^{-5}. We set the weight decay to 0.01 and apply cosine decay with a 3% warmup ratio. Joint training and inference use 1024×1024 images. We set $\lambda_{\text{diff}} = 1$, $\lambda_{\text{aux}} = 0.1$, and the text-region weight in Eq. 4 to $\gamma = 1.2$. Inference uses 40 denoising steps. PAM is applied to the middle half of DiT layers from steps 20 to 30, with attention amplification factor $\alpha = 1.2$.

4.2 Main Results

Quantitative results. On CVTG-2K (Table 1), DuetGen reaches 0.8293 average word accuracy, closely matching the 20B Qwen-Image (Wu et al., 2025a) (0.8301), and performs best on the most crowded five-region split. On LongText-Bench (Table 2), it attains 0.938 accuracy, within 0.005 of Qwen-Image (Wu et al., 2025a) and above all other baselines. These results show competitive text-rendering accuracy on both benchmarks using a 2B planner and a 4B single-stream DiT.

Qualitative results. Figure 3 compares representative multi-region and long-text generations with Janus-Pro (Chen et al., 2025f), FLUX.1 (BlackForest, 2024), DALL-E 3 (Betker et al., 2023), and

Table 1: Quantitative results on CVTG-2K (Du et al., 2025). Word accuracy is grouped by the number of text regions. Best and second-best results are **bold** and underlined.

Method	Word Accuracy ↑					NED ↑	CLIPScore ↑
	2	3	4	5	Avg.		
SD3.5 Large (Esser et al., 2024)	0.7293	0.6825	0.6574	0.5940	0.6548	0.8470	0.7797
FLUX.1 dev (BlackForest, 2024)	0.6089	0.5531	0.4661	0.4316	0.4965	0.6879	0.7401
AnyText (Tuo et al., 2023)	0.0513	0.1739	0.1948	0.2249	0.1804	0.4675	0.7432
TextDiffuser-2 (Chen et al., 2024a)	0.5322	0.3255	0.1787	0.0809	0.2326	0.4353	0.6765
RAG-Diffusion (Chen et al., 2024c)	0.4388	0.3316	0.2116	0.1910	0.2648	0.4498	0.7797
3DIS (Zhou et al., 2024c)	0.4495	0.3959	0.3880	0.3303	0.3813	0.6505	0.7767
TextCrafter (Du et al., 2025)	0.7628	0.7628	0.7406	0.6977	0.7370	0.8679	0.7868
Seedream 3.0 (Gao et al., 2025b)	0.6282	0.5962	0.6043	0.5610	0.5924	0.8537	0.7821
Show-o2 (Xie et al., 2025a)	0.1715	0.1358	0.1193	0.0681	0.1236	0.4152	0.7204
Janus-Pro (Chen et al., 2025f)	0.1791	0.1412	0.1275	0.0736	0.1303	0.4278	0.7329
BLIP3o-Next (Chen et al., 2025c)	0.3821	0.3415	0.3275	0.1736	0.3061	0.4898	0.7319
Qwen-Image (Wu et al., 2025a)	0.8370	**0.8364**	**0.8313**	0.8158	**0.8301**	**0.9116**	**0.7982**
DuetGen	**0.8376**	0.8321	0.8309	**0.8177**	0.8293	0.9113	0.7931

Table 2: Text accuracy on LongText-Bench (Geng et al., 2025). Best and second-best results are **bold** and underlined.

Method	Acc. ↑	Method	Acc. ↑
BLIP3-o (Chen et al., 2025b)	0.021	OmniGen2 (Wu et al., 2025b)	0.561
Show-o2 (Xie et al., 2025a)	0.119	FLUX.1 (BlackForest, 2024)	0.607
Janus-Pro (Chen et al., 2025f)	0.019	TextDiffuser-2 (Chen et al., 2024a)	0.371
Kolors 2.0 (team, 2025)	0.543	HiDream-I1-Full (Cai et al., 2025)	0.543
RAG-Diffusion (Chen et al., 2024c)	0.321	TextCrafter (Du et al., 2025)	0.683
Qwen-Image (Wu et al., 2025a)	**0.943**	DuetGen	0.938

OmniGen2 (Wu et al., 2025b). The examples jointly test text content, regional placement, and visual composition. Across these cases, DuetGen preserves the requested phrases and their intended spatial organization more consistently, while the comparison models exhibit missing or corrupted characters and region-placement errors.

5 DeepFusion Analysis

We analyze DeepFusion along the planning-to-rendering path: whether coordinate-span states encode box geometry, how the joint planning–rendering objective shapes this information, and how the diffusion-side mechanisms execute region-specific conditions. We further compare joint and separate planner training and assess the transfer of the diffusion-side mechanisms across DiT backbones. Unless otherwise stated, all variants follow the settings in Section 4.1.

5.1 Spatial Information at the AR–DiT Interface

5.1.1 Do Coordinate-Span Representations Encode Box Geometry?

We probe the planner's final-layer states associated with each coordinate span before their projection into the DiT condition tokens in Eq. 3. We mean-pool the states within each span and fit a post-hoc ridge regressor to predict (x_1, y_1, x_2, y_2). Evaluation uses held-out prompts containing 1,034 boxes and reports mean R^2, pixel RMSE, and median pixel error at 1024×1024 resolution.

The frozen pretrained planner already encodes linearly decodable box geometry, achieving $R^2 = 0.8578$, 90.5-pixel RMSE, and 44.7-pixel median error in Figure 4(a). Updating the planner with the full joint planning–rendering objective improves these results to $R^2 = 0.9995$, 5.1-pixel RMSE, and 1.9-pixel median error (Figure 4(b)). Thus, the coordinate-span states used at the AR–DiT interface encode box geometry, whose linear readability is substantially improved under the full objective.

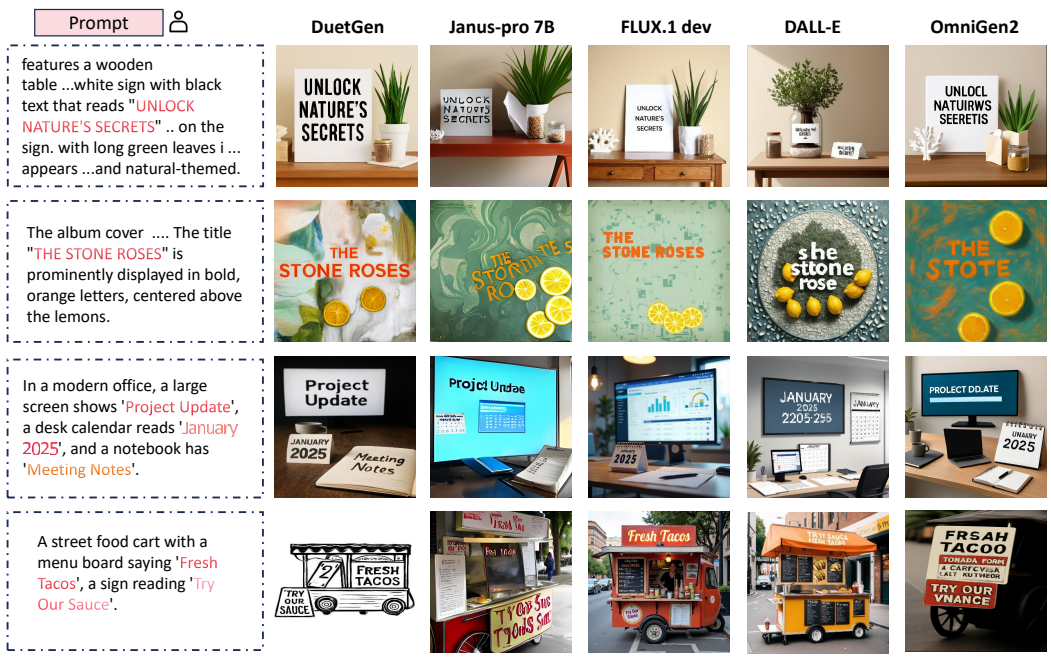


Figure 3: Qualitative comparison on prompts containing multiple text regions and extended text.

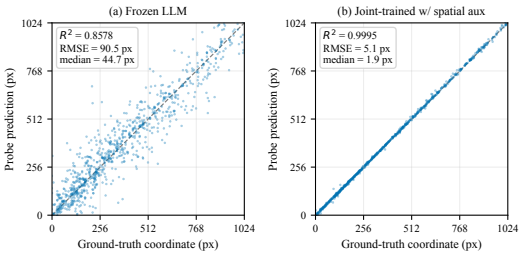


Figure 4: Linear readout of coordinate-span states in the pretrained planner (left) and after optimization with the full objective (right).

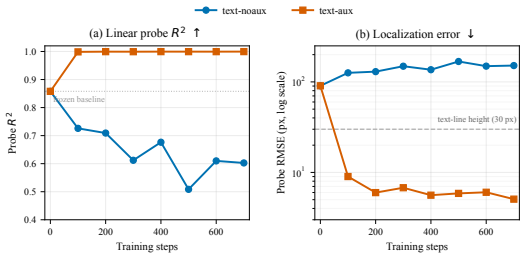


Figure 5: Auxiliary supervision restores spatial readability after diffusion-only degradation.

5.1.2 Is the Diffusion Objective Sufficient?

We isolate the effects of different objectives on planner optimization while keeping DiT training fixed. The planner is either frozen or updated by $\mathcal{L}_{\text{diff}}$, $\mathcal{L}_{\text{diff}} + \mathcal{L}_{\text{plan}}$, or the full joint objective. Each checkpoint is evaluated with an independently fitted linear probe.

Diffusion-only training degrades spatial readability, reducing R^2 from 0.8578 to 0.6026 and increasing RMSE from 90.5 to 151.5 pixels. Adding autoregressive plan supervision provides only marginal recovery ($R^2 = 0.6129$, RMSE $= 147.3$). Adding auxiliary coordinate supervision instead improves the probe to $R^2 = 0.9995$ and 5.1-pixel RMSE.

The same pattern appears in planning–rendering consistency (PRC), measured as the mean IoU between planned and rendered text boxes. On the four-region CVTG-2K subset (Du et al., 2025), PRC decreases from 0.4892 with a frozen planner to 0.3121 under diffusion-only training, while adding the plan loss recovers it only to 0.3469 (Table 3). The full objective raises PRC to 0.9176. These results connect auxiliary coordinate supervision to both the spatial readability of coordinate-span states and the accuracy with which the DiT executes the generated layout.

Table 3: Planner-update effects on spatial readability and generation on the four-region CVTG-2K subset.

Planner update	Spatial probe			Generation		
	R^2 ↑	RMSE$_{px}$ ↓	Med.$_{px}$ ↓	Acc.↑	PRC↑	CLIP↑
Frozen	0.8578	90.5	44.7	0.7032	0.4892	0.6987
$\mathcal{L}_{\text{diff}}$	0.6026	151.5	66.4	0.6788	0.3121	0.7011
$\mathcal{L}_{\text{diff}} + \mathcal{L}_{\text{plan}}$	0.6129	147.3	65.7	0.6832	0.3469	0.7187
$\mathcal{L}_{\text{diff}} + \mathcal{L}_{\text{plan}} + \mathcal{L}_{\text{aux}}$	**0.9995**	**5.1**	**1.9**	**0.8309**	**0.9176**	**0.7925**

Table 4: Jointly optimized 2B planner versus same-size and larger separately trained planners. Lower OR indicates fewer conflicts among planned boxes.

Planner	Train.	Params	Acc.↑	PRC↑	OR↓
DuetGen planner	Joint	2B	**0.8309**	**0.9176**	**0.063**
Qwen3.5-2B	Sep. SFT	2B	0.6341	0.6592	0.116
Llama3.1-8B	Sep. SFT	8B	0.6783	0.6624	0.129
Qwen3.5-9B	Sep. SFT	9B	0.6936	0.7189	0.114

5.1.3 Does Spatial Readability Depend on Pretrained Digit Tokens?

Having established the role of auxiliary coordinate supervision, we test whether its effect depends on the pretrained numerical structure of digit tokens. We compare the digit serialization used by DuetGen, such as $[235, 45, 787, 71]$, with a newly initialized coordinate-bin vocabulary containing 64 tokens per axis. Each bin covers 16 pixels at 1024×1024 resolution; for example, the same box is represented as $\langle \text{xbin14} \rangle \langle \text{ybin2} \rangle \langle \text{xbin49} \rangle \langle \text{ybin4} \rangle$.

Under the same coordinate-supervised setup, the randomly initialized bin tokens become linearly readable after approximately 150–200 steps and converge to a 6.4-pixel RMSE, close to the 5.1 pixels obtained with pretrained digits (Figure 6). Thus, linearly readable spatial representations do not require pretrained digit tokens; digit serialization mainly accelerates convergence and yields slightly lower final error.

5.1.4 Can Planner Scale Replace Joint Optimization?

We test whether planner scale can replace joint optimization. We compare the jointly optimized Qwen3.5-2B planner with separately fine-tuned Qwen3.5-2B, Llama3.1-8B (Meta, 2024), and Qwen3.5-9B (Qwen Team, 2026) planners under matched layout supervision, renderer initialization, and data. Sep. SFT freezes the planner and trains only the mapper–DiT with $\mathcal{L}_{\text{diff}}$; joint training also updates the planner with it. On the four-region CVTG-2K subset, we report word accuracy, PRC, and overlap rate (OR).

The same-size separate control reaches only 0.6341 accuracy, 0.6592 PRC, and 0.116 OR (Table 4); even the separate 9B planner remains below the jointly optimized 2B planner. Thus, neither separate training nor planner scaling recovers the planning–rendering alignment achieved by joint optimization.

5.2 Region-Specific Execution

Building on the spatial analysis above, we evaluate two diffusion-side mechanisms for region-specific execution: text-region-weighted diffusion training and PAM at inference.

Text-region-weighted diffusion learning. Table 5(a) varies the text-region weight γ on the Styled-TextSynth evaluation subset of TextAtlasEval (Wang et al., 2025b), with PAM enabled. Increasing γ from 1.0 to 1.2 reduces FID from 74.15 to 68.70 and raises OCR accuracy from 48.32 to 69.93. Larger weights reverse part of these gains, with $\gamma = 1.2$ achieving the best result across all three metrics.

Phase-Aware Attention Modulation. PAM selectively strengthens attention from image tokens inside each planned box to the region-matched bbox and content tokens, restricting the modulation to the middle DiT layers and denoising phase. Figure 7 qualitatively shows that PAM more faithfully renders the requested strings across multiple text regions.

On the same evaluation subset, with $\gamma = 1.2$, PAM increases OCR accuracy from 61.71 to 69.93, reduces FID from 72.03 to 68.70, and improves CLIPScore from 0.3168 to 0.3271 (Table 5(a)). Table 5(c) further localizes these gains: the middle denoising phase and middle layer range perform best in their respective sweeps. Applying PAM to all steps or layers retains part of the improvement but underperforms the localized setting. Increasing α beyond 1.2 similarly reverses part of the gain.

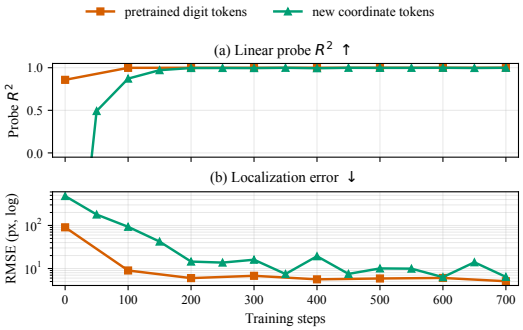


Figure 6: Spatial readability of pretrained digit and newly initialized coordinate-bin tokens.



Figure 7: Qualitative comparison with and without PAM.

Table 5: Diffusion-side ablations on the StyledTextSynth evaluation subset of TextAtlasEval: (a) text-region weighting and PAM, (b) backbone transfer, and (c) PAM design. CS and Acc. denote CLIPScore and OCR accuracy.

(a) Main DiT ablations

Study	Setting	FID↓	CS↑	Acc.↑
Region wt.	$\gamma = 1.0$	74.15	0.2640	48.32
	$\gamma = \mathbf{1.2}$	**68.70**	**0.3271**	**69.93**
	$\gamma = 1.5$	73.50	0.2880	54.10
	$\gamma = 2.0$	77.80	0.2750	55.24
PAM	PAM off	72.03	0.3168	61.71
	PAM on	**68.70**	**0.3271**	**69.93**

(b) Transfer across diffusion backbones

Region wt.	PAM	PixArt-Σ			SD3		
		FID↓	CS↑	Acc.↑	FID↓	CS↑	Acc.↑
×	×	82.83	0.2764	31.42	84.2	0.281	49.22
✓	×	82.3	0.283	43.27	78.5	0.289	57.23
×	✓	85.1	0.278	41.99	81.7	0.284	54.24
✓	✓	**71.4**	**0.297**	**48.63**	**70.1**	**0.301**	**59.55**

(c) PAM design ablations

Denoising phase				Layer range				Amplification			
Setting	FID↓	CS↑	Acc.↑	Setting	FID↓	CS↑	Acc.↑	α	FID↓	CS↑	Acc.↑
Early	74.00	0.3195	64.30	Early half	72.40	0.3203	63.60	1.0	72.03	0.3168	61.71
Middle	**68.70**	**0.3271**	**69.93**	Middle half	**68.70**	**0.3271**	**69.93**	1.1	69.90	0.3226	66.40
Late	71.60	0.3192	62.80	Late half	70.90	0.3216	64.80	1.2	**68.70**	**0.3271**	**69.93**
All	71.30	0.3251	68.10	All layers	70.60	0.3249	67.20	1.4	72.50	0.3238	67.90

These results support PAM as a localized attention modulation rather than uniform amplification throughout denoising.

5.3 DiT Backbone Transfer

We evaluate the diffusion-side mechanisms within DeepFusion using PixArt-Σ (Chen et al., 2024b) and SD3 (Esser et al., 2024) as alternative DiT backbones, with the same StyledTextSynth evaluation subset and protocol. For each backbone, we train models with and without text-region weighting and evaluate both checkpoints with and without PAM. Both mechanisms improve CLIPScore and OCR accuracy across the two architectures, while their combination achieves the best result on all three metrics, reaching OCR accuracies of 48.63 on PixArt-Σ and 59.55 on SD3 (Table 5(b)). These consistent gains show that text-region weighting and PAM transfer across distinct DiT architectures.

6 Conclusion

We presented DuetGen, an autonomous visual text generator that makes spatial plans executable through joint autoregressive–diffusion learning. DeepFusion aligns planner states with continuous rendering through planning, spatial, and text-region supervision; PAM strengthens region-specific execution at inference. Experiments show improved planning–rendering consistency and image quality, with a 2B planner and 4B DiT approaching larger models in text accuracy. These findings support joint learning as a basis for faithful plan execution in visual text generation.

7 AI USE STATEMENT

Generative AI tools were used to assist with the implementation and debugging of research code, the formulation and presentation of mathematical claims and proofs, and the drafting and language editing of portions of the manuscript. The authors reviewed and validated all AI-assisted code, mathematical arguments, and manuscript text. Generative AI tools were not used for literature retrieval or discovery, or for generating synthetic training or evaluation datasets. The authors take full responsibility for the final content of this work.

REFERENCES

James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf*, 2(3):8, 2023.

BlackForest. Black forest labs; frontier ai lab, 2024. URL `https://blackforestlabs.ai/`.

Minwoo Byeon, Beomhee Park, Haecheon Kim, Sungjun Lee, Woonhyuk Baek, and Saehoon Kim. COYO-700M: Image-text pair dataset. `https://github.com/kakaobrain/coyo-dataset`, 2022.

Qi Cai, Jingwen Chen, Yang Chen, Yehao Li, Fuchen Long, Yingwei Pan, Zhaofan Qiu, Yiheng Zhang, Fengbin Gao, Peihan Xu, et al. Hidream-i1: A high-efficient image generative foundation model with sparse diffusion transformer. *arXiv preprint arXiv:2505.22705*, 2025.

Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3558–3568, 2021.

Jingye Chen, Yupan Huang, Tengchao Lv, Lei Cui, Qifeng Chen, and Furu Wei. Textdiffuser: Diffusion models as text painters. *Advances in Neural Information Processing Systems*, 36: 9353–9387, 2023.

Jingye Chen, Yupan Huang, Tengchao Lv, Lei Cui, Qifeng Chen, and Furu Wei. Textdiffuser-2: Unleashing the power of language models for text rendering. In *European Conference on Computer Vision*, pp. 386–402. Springer, 2024a.

Jingye Chen, Zhaowen Wang, Nanxuan Zhao, Li Zhang, Difan Liu, Jimei Yang, and Qifeng Chen. Rethinking layered graphic design generation with a top-down approach. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16861–16870, 2025a.

Jiuhai Chen, Zhiyang Xu, Xichen Pan, Yushi Hu, Can Qin, Tom Goldstein, Lifu Huang, Tianyi Zhou, Saining Xie, Silvio Savarese, et al. Blip3-o: A family of fully open unified multimodal models-architecture, training and dataset. *arXiv preprint arXiv:2505.09568*, 2025b.

Jiuhai Chen, Le Xue, Zhiyang Xu, Xichen Pan, Shusheng Yang, Can Qin, An Yan, Honglu Zhou, Zeyuan Chen, Lifu Huang, et al. Blip3o-next: Next frontier of native image generation. *arXiv preprint arXiv:2510.15857*, 2025c.

Junsong Chen, Chongjian Ge, Enze Xie, Yue Wu, Lewei Yao, Xiaozhe Ren, Zhongdao Wang, Ping Luo, Huchuan Lu, and Zhenguo Li. PixArt-σ: Weak-to-strong training of diffusion transformer for 4k text-to-image generation. In *European Conference on Computer Vision*, pp. 74–91. Springer, 2024b. doi: 10.1007/978-3-031-73411-3_5.

Junwen Chen, Heyang Jiang, Yanbin Wang, Keming Wu, Ji Li, Chao Zhang, Keiji Yanai, Dong Chen, and Yuhui Yuan. Prismlayers: Open data for high-quality multi-layer transparent image generative models. *arXiv preprint arXiv:2505.22523*, 2025d.

SiXiang Chen, Jianyu Lai, Jialin Gao, Tian Ye, Haoyu Chen, Hengyu Shi, Shitong Shao, Yunlong Lin, Song Fei, Zhaohu Xing, et al. Postercraft: Rethinking high-quality aesthetic poster generation in a unified framework. *arXiv preprint arXiv:2506.10741*, 2025e.

Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025f.

Zhennan Chen, Yajie Li, Haofan Wang, Zhibo Chen, Zhengkai Jiang, Jun Li, Qian Wang, Jian Yang, and Ying Tai. Region-aware text-to-image generation via hard binding and soft refinement. *arXiv preprint arXiv:2411.06558*, 2024c.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, et al. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

Karan Desai, Gaurav Kaul, Zubin Aysola, and Justin Johnson. Redcaps: Web-curated image-text data created by the people, for the people. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, 2021.

Nikai Du, Zhennan Chen, Shan Gao, Zhizhou Chen, Xi Chen, Zhengkai Jiang, Jian Yang, and Ying Tai. Textcrafter: Accurately rendering multiple texts in complex visual scenes. *arXiv preprint arXiv:2503.23461*, 2025.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

Yifan Gao, Zihang Lin, Chuanbin Liu, Min Zhou, Tiezheng Ge, Bo Zheng, and Hongtao Xie. Postermaker: Towards high-quality product poster generation with accurate text rendering. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 8083–8093, 2025a.

Yu Gao, Lixue Gong, Qiushan Guo, Xiaoxia Hou, Zhichao Lai, Fanshi Li, Liang Li, Xiaochen Lian, Chao Liao, Liyang Liu, et al. Seedream 3.0 technical report. *arXiv preprint arXiv:2504.11346*, 2025b.

Zigang Geng, Yibing Wang, Yeyao Ma, Chen Li, Yongming Rao, Shuyang Gu, Zhao Zhong, Qinglin Lu, Han Hu, Xiaosong Zhang, et al. X-omni: Reinforcement learning makes discrete autoregressive image generative models great again. *arXiv preprint arXiv:2507.22058*, 2025.

Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. *Advances in Neural Information Processing Systems*, 36: 52132–52152, 2023.

Runze He, Bo Cheng, Yuhang Ma, Qingxiang Jia, Shanyuan Liu, Ao Ma, Xiaoyu Wu, Liebucha Wu, Dawei Leng, and Yuhui Yin. Plangen: Towards unified layout planning and image generation in auto-regressive vision language models. *arXiv preprint arXiv:2503.10127*, 2025.

Xiwei Hu, Haokun Chen, Zhongqi Qi, Hui Zhang, Dexiang Hong, Jie Shao, and Xinglong Wu. Dreamposter: A unified framework for image-conditioned generative poster design. *arXiv preprint arXiv:2507.04218*, 2025.

Jiabao Ji, Guanhua Zhang, Zhaowen Wang, Bairu Hou, Zhifei Zhang, Brian Price, and Shiyu Chang. Improving diffusion models for scene text editing with dual encoders. *arXiv preprint arXiv:2304.05568*, 2023.

Bowen Jiang, Yuan Yuan, Xinyi Bai, Zhuoqun Hao, Alyson Yin, Yaojie Hu, Wenyu Liao, Lyle Ungar, and Camillo J Taylor. Controltext: Unlocking controllable fonts in multilingual text rendering without font annotations. *arXiv preprint arXiv:2502.10999*, 2025.

Caoshuo Li, Zengmao Ding, Xiaobin Hu, Bang Li, Donghao Luo, AndyPian Wu, Chaoyang Wang, Chengjie Wang, Taisong Jin, Seven Shu, et al. Oraclefusion: Assisting the decipherment of oracle bone script with structurally constrained semantic typography. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 19893–19902, 2025a.

Chao Li, Chen Jiang, Xiaolong Liu, Jun Zhao, and Guoxin Wang. Joytype: A robust design for multilingual visual text creation. *arXiv preprint arXiv:2409.17524*, 2024.

Mingcheng Li, Xiaolu Hou, Ziyang Liu, Dingkang Yang, Ziyun Qian, Jiawei Chen, Jinjie Wei, Yue Jiang, Qingyao Xu, and Lihua Zhang. Mccd: Multi-agent collaboration-based compositional diffusion for complex text-to-image generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 13263–13272, 2025b.

Tianyi Liang, Jiangqi Liu, Yifei Huang, Shiqi Jiang, Jianshen Shi, Changbo Wang, and Chenhui Li. Textcengen: Attention-guided text-centric background adaptation for text-to-image generation. *arXiv preprint arXiv:2404.11824*, 2024.

Yiqi Lin, Conghui He, Alex Jinpeng Wang, Bin Wang, Weijia Li, and Mike Zheng Shou. Parrot captions teach clip to spot text. In *European Conference on Computer Vision*, pp. 368–385. Springer, 2024.

Zeyu Liu, Weicong Liang, Zhanhao Liang, Chong Luo, Ji Li, Gao Huang, and Yuhui Yuan. Glyph-byt5: A customized text encoder for accurate visual text rendering. In *European Conference on Computer Vision*, pp. 361–377. Springer, 2024a.

Zeyu Liu, Weicong Liang, Yiming Zhao, Bohan Chen, Lin Liang, Lijuan Wang, Ji Li, and Yuhui Yuan. Glyph-byt5-v2: A strong aesthetic baseline for accurate multilingual visual text rendering. *arXiv preprint arXiv:2406.10208*, 2024b.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Shuo Lu, Yanyin Chen, Wei Feng, Jiahao Fan, Fengheng Li, Zheng Zhang, Jingjing Lv, Junjie Shen, Ching Law, and Jian Liang. Uni-layout: Integrating human feedback in unified layout generation and evaluation. *arXiv preprint arXiv:2508.02374*, 2025.

Chuofan Ma, Yi Jiang, Junfeng Wu, Jihan Yang, Xin Yu, Zehuan Yuan, Bingyue Peng, and Xiaojuan Qi. Unitok: A unified tokenizer for visual generation and understanding. *arXiv preprint arXiv:2502.20321*, 2025a.

Jian Ma, Mingjun Zhao, Chen Chen, Ruichen Wang, Di Niu, Haonan Lu, and Xiaodong Lin. Glyphdraw: Seamlessly rendering text with intricate spatial structures in text-to-image generation. *arXiv preprint arXiv:2303.17870*, 2023.

Jian Ma, Yonglin Deng, Chen Chen, Nanyang Du, Haonan Lu, and Zhenyu Yang. Glyphdraw2: Automatic generation of complex glyph posters with diffusion models and large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 5955–5963, 2025b.

Dongxing Mao, Yilin Wang, Linjie Li, Zhengyuan Yang, and Alex Jinpeng Wang. TextGround4M: A prompt-aligned dataset for layout-aware text rendering. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(10):7918–7926, 2026. doi: 10.1609/aaai.v40i10.37736. URL `https://ojs.aaai.org/index.php/AAAI/article/view/37736`.

Meta. Llama-3.1-8b-instruct. `https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct`, 2024.

Yuyang Peng, Shishi Xiao, Keming Wu, Qisheng Liao, Bohan Chen, Kevin Lin, Danqing Huang, Ji Li, and Yuhui Yuan. Bizgen: Advancing article-level visual text rendering for infographics generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 23615–23624, 2025.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL `https://qwen.ai/blog?id=qwen3.5`.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. Zero: Memory optimizations toward training trillion parameter models. In *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis*, pp. 1–16. IEEE, 2020.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022.

Wenda Shi, Yiren Song, Zihan Rao, Dengming Zhang, Jiaming Liu, and Xingxing Zou. Wordcon: Word-level typography control in scene text rendering. *arXiv preprint arXiv:2506.21276*, 2025a.

Wenda Shi, Yiren Song, Dengming Zhang, Jiaming Liu, and Xingxing Zou. Fonts: Text rendering with typography and style controls. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 18463–18474, 2025b.

Krishna Srinivasan, Karthik Raman, Jiecao Chen, Michael Bendersky, and Marc Najork. Wit: Wikipedia-based image text dataset for multimodal multilingual machine learning. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2443–2449, 2021.

Keqiang Sun, Junting Pan, Yuying Ge, Hao Li, Haodong Duan, Xiaoshi Wu, Renrui Zhang, Aojun Zhou, Zipeng Qin, Yi Wang, et al. Journeydb: A benchmark for generative image understanding. *Advances in Neural Information Processing Systems*, 36:49659–49678, 2023.

Kuaishou Kolors team. Kolors2.0. https://app.klingai.com/cn/, 2025.

Yuxiang Tuo, Wangmeng Xiang, Jun-Yan He, Yifeng Geng, and Xuansong Xie. Anytext: Multilingual visual text generation and editing. *arXiv preprint arXiv:2311.03054*, 2023.

Yuxiang Tuo, Yifeng Geng, and Liefeng Bo. Anytext2: Visual text generation and editing with customizable attributes. *arXiv preprint arXiv:2411.15245*, 2024.

Alex Jinpeng Wang, Linjie Li, Zhengyuan Yang, Lijuan Wang, and Min Li. Beyond words: Advancing long-text image generation via multimodal autoregressive models. *arXiv preprint arXiv:2503.20198*, 2025a.

Alex Jinpeng Wang, Dongxing Mao, Jiawei Zhang, Weiming Han, Zhuobai Dong, Linjie Li, Yiqi Lin, Zhengyuan Yang, Libo Qin, Fuwei Zhang, et al. Textatlas5m: A large-scale dataset for dense text image generation. *arXiv preprint arXiv:2502.07870*, 2025b.

Haofan Wang, Yujia Xu, Yimeng Li, Junchen Li, Chaowei Zhang, Jing Wang, Kejia Yang, and Zhibo Chen. Reptext: Rendering visual text via replicating. *arXiv preprint arXiv:2504.19724*, 2025c.

Yibin Wang, Weizhong Zhang, Honghui Xu, and Cheng Jin. Dreamtext: High fidelity scene text synthesis. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 28555–28563, 2025d.

Zhendong Wang, Jianmin Bao, Shuyang Gu, Dong Chen, Wengang Zhou, and Houqiang Li. Design-diffusion: High-quality text-to-design image generation with diffusion models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 20906–20915, 2025e.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-image technical report. *arXiv preprint arXiv:2508.02324*, 2025a.

Chenyuan Wu, Pengfei Zheng, Ruiran Yan, Shitao Xiao, Xin Luo, Yueze Wang, Wanli Li, Xiyan Jiang, Yexin Liu, Junjie Zhou, et al. Omnigen2: Exploration to advanced multimodal generation. *arXiv preprint arXiv:2506.18871*, 2025b.

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

Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation. *arXiv preprint arXiv:2408.12528*, 2024.

Jinheng Xie, Zhenheng Yang, and Mike Zheng Shou. Show-o2: Improved native unified multimodal models. *arXiv preprint arXiv:2506.15564*, 2025a.

Yu Xie, Jielei Zhang, Pengyu Chen, Ziyue Wang, Weihang Wang, Longwen Gao, Peiyi Li, Huyang Sun, Qiang Zhang, Qian Qiao, et al. Textflux: An ocr-free dit model for high-fidelity multilingual scene text synthesis. *arXiv preprint arXiv:2505.17778*, 2025b.

Yukang Yang, Dongnan Gui, Yuhui Yuan, Weicong Liang, Haisong Ding, Han Hu, and Kai Chen. Glyphcontrol: Glyph conditional control for visual text generation. *Advances in Neural Information Processing Systems*, 36:44050–44066, 2023.

Lijun Yu, José Lezama, Nitesh B Gundavarapu, Luca Versari, Kihyuk Sohn, David Minnen, Yong Cheng, Vighnesh Birodkar, Agrim Gupta, Xiuye Gu, et al. Language model beats diffusion–tokenizer is key to visual generation. *arXiv preprint arXiv:2310.05737*, 2023.

Z-Image Team. Z-image: An efficient image generation foundation model with single-stream diffusion transformer. *arXiv preprint arXiv:2511.22699*, 2025.

Hui Zhang, Dexiang Hong, Yitong Wang, Jie Shao, Xinglong Wu, Zuxuan Wu, and Yu-Gang Jiang. Creatilayout: Siamese multimodal diffusion transformer for creative layout-to-image generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 18487–18497, 2025a.

Hui Zhang, Dexiang Hong, Maoke Yang, Yutao Cheng, Zhao Zhang, Jie Shao, Xinglong Wu, Zuxuan Wu, and Yu-Gang Jiang. Creatidesign: A unified multi-conditional diffusion transformer for creative graphic design. *arXiv preprint arXiv:2505.19114*, 2025b.

Lingjun Zhang, Xinyuan Chen, Yaohui Wang, Yue Lu, and Yu Qiao. Brush your text: Synthesize any scene text on images via diffusion model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 7215–7223, 2024.

Qilong Zhangli, Jindong Jiang, Di Liu, Licheng Yu, Xiaoliang Dai, Ankit Ramchandani, Guan Pang, Dimitris N Metaxas, and Praveen Krishnan. Layout-agnostic scene text image synthesis with diffusion models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7496–7506. IEEE Computer Society, 2024.

Lirui Zhao, Yue Yang, Kaipeng Zhang, Wenqi Shao, Yuxin Zhang, Yu Qiao, Ping Luo, and Rongrong Ji. Diffagent: Fast and accurate text-to-image api selection with large language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6390–6399, 2024.

Shitian Zhao, Qilong Wu, Xinyue Li, Bo Zhang, Ming Li, Qi Qin, Dongyang Liu, Kaipeng Zhang, Hongsheng Li, Yu Qiao, et al. Lex-art: Rethinking text generation via scalable high-quality data synthesis. *arXiv preprint arXiv:2503.21749*, 2025.

Yiming Zhao and Zhouhui Lian. Udifftext: A unified framework for high-quality text synthesis in arbitrary images via character-aware diffusion models. In *European conference on computer vision*, pp. 217–233. Springer, 2024.

Dingcheng Zhen, Qian Qiao, Xu Zheng, Tan Yu, Kangxi Wu, Ziwei Zhang, Siyuan Liu, Shunshun Yin, and Ming Tao. Marrying autoregressive transformer and diffusion with multi-reference autoregression. *arXiv preprint arXiv:2506.09482*, 2025.

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

Chunting Zhou, Lili Yu, Arun Babu, Kushal Tirumala, Michihiro Yasunaga, Leonid Shamis, Jacob Kahn, Xuezhe Ma, Luke Zettlemoyer, and Omer Levy. Transfusion: Predict the next token and diffuse images with one multi-modal model. *arXiv preprint arXiv:2408.11039*, 2024a.

Dewei Zhou, You Li, Fan Ma, Xiaoting Zhang, and Yi Yang. Migc: Multi-instance generation controller for text-to-image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6818–6828, 2024b.

Dewei Zhou, Ji Xie, Zongxin Yang, and Yi Yang. 3dis: Depth-driven decoupled instance synthesis for text-to-image generation. *arXiv preprint arXiv:2410.12669*, 2024c.

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*, 2025.

Shijie Zhou, Ruiyi Zhang, Yufan Zhou, and Changyou Chen. A high-quality text-rich image instruction tuning dataset via hybrid instruction generation. *arXiv preprint arXiv:2412.16364*, 2024d.

A DiT Architecture and Generic Pretraining

Architecture. Our renderer is a single-stream diffusion transformer with approximately 4B parameters, pretrained from random initialization. The mapper projects the planner states to the DiT width. Noised-image and condition tokens form the joint sequence $[\mathbf{Z}_t; \mathbf{C}]$ in Eq. 3, which is processed by shared attention and feed-forward layers throughout the network. We use the pretrained VAE released with Z-Image (Z-Image Team, 2025) for image encoding and decoding and keep its parameters frozen. During training, noise is added to the encoded image latents; the DiT operates on patch tokens of these noised latents. At inference, generation starts from latent noise, and the VAE decoder maps the denoised latents to an image. The stated parameter count refers to the DiT alone.

Image-prior initialization. We first train the DiT for 200K steps on ImageNet-1K (Deng et al., 2009) to learn an initial image prior before generic text-to-image pretraining.

Generic text-to-image pretraining. We pretrain the DiT on a fixed mixture of 15M image–caption pairs at 512×512 resolution. The mixture is constructed from Conceptual 12M (CC12M) (Changpinyo et al., 2021), high-quality subsets of LAION (Schuhmann et al., 2022) and COYO (Byeon et al., 2022), RedCaps (Desai et al., 2021), JourneyDB (Sun et al., 2023), and WIT (Srinivasan et al., 2021). Table 6 reports the exact composition. This stage provides a general image–text prior before the joint DeepFusion training described in Section 3.2.

Optimization. For generic text-to-image pretraining, we train on 16 NVIDIA A800 GPUs with a global batch size of 256 for 240K optimizer steps. We use AdamW with a peak learning rate of 1×10^{-4} and weight decay of 0.01. The learning rate is linearly warmed up over the first 7.2K steps (3% of training) and then decayed with a cosine schedule. The joint stage subsequently optimizes the planner, mapper, DiT, and auxiliary head with the objective in Eq. 6; PAM is applied only during inference. Section 5 evaluates the proposed components through controlled comparisons.

Table 6: Composition of the 15M-pair generic text-to-image pretraining mixture. The high-quality LAION and COYO subsets are sampled before pretraining.

Source	Pairs	Share	Primary coverage
CC12M	4.0M	26.7%	Real-world images and long-tail concepts
LAION (high-quality subset)	3.5M	23.3%	Visual concepts, styles, and scenes
COYO (high-quality subset)	2.5M	16.7%	Web scenes and contemporary imagery
RedCaps	2.0M	13.3%	Natural user-authored descriptions
JourneyDB	2.0M	13.3%	Aesthetic synthetic images and detailed prompts
WIT	1.0M	6.7%	Entity, cultural, and limited multilingual coverage
Total	**15.0M**	**100.0%**	

B Human Evaluation and General Generation

B.1 Human Evaluation

Automatic metrics do not fully capture whether rendered text is visually integrated with its surrounding scene. We therefore conduct a double-blind human evaluation against AnyText (Tuo et al., 2023), TextDiffuser-2 (Chen et al., 2024a), and FLUX.1 dev (BlackForest, 2024). We randomly sample 50 prompts from CVTG-2K without cherry-picking and recruit 20 participants, including graphic designers and general users. Model identities are anonymized. Participants score each image from 1 to 10 for visual aesthetics, text quality, and spatial harmony.

As shown in Table 7, DuetGen receives the highest scores in text quality and spatial harmony. These results complement the automatic evaluation by measuring whether the planned text is both readable and compositionally compatible with the generated scene.

Table 7: Human evaluation on 50 randomly sampled CVTG-2K prompts. Scores range from 1 to 10; higher is better.

Method	Vis. Aes.↑	Text Qual.↑	Spatial Harm.↑
AnyText	7.12	7.85	7.43
TextDiffuser-2	7.45	8.21	7.92
FLUX.1 dev	**8.92**	8.64	8.15
DuetGen	8.75	**9.12**	**8.95**

B.2 General Image Generation

We additionally evaluate general text-to-image generation after joint training. Table 8 reports the GenEval (Ghosh et al., 2023) results across single-object, two-object, counting, color, position, and color-attribution tasks, with an overall score of 0.91. These results characterize the final model; a comparison with the pretraining checkpoint would be needed to measure changes in general generation capability during joint training.

Table 8: GenEval (Ghosh et al., 2023) results. Sng. Obj.: single object; Tw. Obj.: two objects; Cntng: counting; Clrs: colors; Pstn: position; Clr Attr.: color attribution.

Model	Sng. Obj.	Tw. Obj.	Cntng	Clrs	Pstn	Clr Attr.	Overall
DuetGen	0.99	0.95	0.88	0.90	0.92	0.79	**0.91**

C Text-Region-Weighted Flow-Matching Loss

Algorithm 1 details the minibatch computation of the text-region-weighted objective in Eq. 4. For each training image, we rasterize its normalized reference boxes onto the latent grid and assign weight γ to every covered patch while leaving all other patches at unit weight. The resulting map reweights the squared flow-matching residual, whose target velocity is $\mathbf{v} = \boldsymbol{\epsilon} - \mathbf{z}_0$. Overlapping boxes do not compound the weight: a patch covered by one or more boxes always receives weight γ.

Here, $\mathbf{u}$ indexes a spatial location on the $H \times W$ latent grid, and the ℓ_2 norm sums the residual over its C channels. The floor/ceiling conversion includes every latent patch intersected by a reference box. Consequently, setting $\gamma = 1$ exactly recovers the standard flow-matching objective, whereas $\gamma > 1$ increases the contribution of the union of text regions without removing supervision from the rest of the image.

D Counterfactual Bbox-Following Test

We use a controlled intervention to test whether the DiT follows the planner's bbox representation. For a prompt p, we first obtain its clean bbox–content plan $s = \{(b_i, c_i)\}_{i=1}^{K}$ and its condition tokens $\mathbf{C}$. We then construct a counterfactual plan $\tilde{s} = \{(\tilde{b}_i, c_i)\}_{i=1}^{K}$ by changing only the box locations. Each $\tilde{b}_i$ preserves the width and height of b_i, remains fully inside the image, and does not overlap another counterfactual box. Candidate locations that violate these validity constraints are rejected. The relative target displacement is $r = \frac{1}{K}\sum_i \frac{\|\mathrm{ctr}(\tilde{b}_i) - \mathrm{ctr}(b_i)\|_2}{\sqrt{w_i^2 + h_i^2}} = 0.35$, measured in units of the original bbox diagonal. We run the planner with forced decoding on $\tilde{s}$ and replace only the coordinate-span condition tokens in $\mathbf{C}$ with their forced-plan counterparts. The prompt condition, text-content condition tokens, DiT weights, and sampling seed remain fixed.

The test directly measures whether the rendered text regions follow the injected target boxes. Let $\hat{b}_i$ be the OCR-detected box in the counterfactual generation. *Target IoU* is the mean $\mathrm{IoU}(\hat{b}_i, \tilde{b}_i)$. We

Algorithm 1: Text-Region-Weighted Flow-Matching Loss

Input: Predictions $\mathbf{v}_\phi \in \mathbb{R}^{B\times C\times H\times W}$,
target velocities $\mathbf{v} = \boldsymbol{\epsilon} - \mathbf{z}_0 \in \mathbb{R}^{B\times C\times H\times W}$,
normalized reference boxes $\{\mathcal{B}_n\}_{n=1}^{B}$,
text-region weight $\gamma \geq 1$
Output: Minibatch loss $\widehat{\mathcal{L}}_{\text{diff}}$

for $n \leftarrow 1$ **to** B **do**
  // Start with the standard flow-matching weight at every latent patch
  $\mathbf{W}_n \leftarrow \mathbf{1}^{H\times W}$
  // Rasterize each valid normalized box onto the latent grid
  foreach $b = (x_1, y_1, x_2, y_2) \in \mathcal{B}_n$ **do**
    if $0 \leq x_1 < x_2 \leq 1$ *and* $0 \leq y_1 < y_2 \leq 1$ **then**
      $j_1 \leftarrow \lfloor W x_1 \rfloor$, $j_2 \leftarrow \min(W, \lceil W x_2 \rceil)$
      $i_1 \leftarrow \lfloor H y_1 \rfloor$, $i_2 \leftarrow \min(H, \lceil H y_2 \rceil)$
      // Use a union mask, so overlaps remain weighted by γ
      $\mathbf{W}_n[i_1 : i_2, j_1 : j_2] \leftarrow \gamma$
    end
  end
  // The channelwise residual norm gives one scalar error per patch
  $\ell_n \leftarrow \sum_{\mathbf{u}} \mathbf{W}_n(\mathbf{u}) \, \|\mathbf{v}_{\phi,n}(\mathbf{u}) - \mathbf{v}_n(\mathbf{u})\|_2^2$
end
$\widehat{\mathcal{L}}_{\text{diff}} \leftarrow \frac{1}{B} \sum_{n=1}^{B} \ell_n$
return $\widehat{\mathcal{L}}_{\text{diff}}$

Table 9: Counterfactual bbox-following results on the four-region CVTG-2K subset. The forced-shift plan preserves box size, stays in bounds, and contains no overlapping pair; only coordinate-span condition tokens differ from the clean run.

Condition	Target shift r	Target IoU↑	Center Error↓	Word Acc.↑	CLIPScore↑
Clean self-generated plan	0.00	0.9176	0.068	0.8309	0.7925
Forced valid-shift plan	0.35	0.9042	0.074	0.8295	0.7941

define *center error* relative to the diagonal of the target box,

$$E_{\text{ctr}} = \frac{1}{K} \sum_{i=1}^{K} \frac{\left\| \text{ctr}(\hat{b}_i) - \text{ctr}(\tilde{b}_i) \right\|_2}{\sqrt{\tilde{w}_i^2 + \tilde{h}_i^2}}, \tag{8}$$

where $\text{ctr}(b)$ denotes the box center and $(\tilde{w}_i, \tilde{h}_i)$ are the width and height of $\tilde{b}_i$. Thus, $E_{\text{ctr}} = 0$ is exact location following, while $E_{\text{ctr}} = 1$ means that the rendered center is one target-box diagonal away. We use the clean-plan center error as the internal reference: a forced-shift error close to the clean value, together with high Target IoU, shows that the DiT executes the counterfactual bbox with comparable spatial precision. Word Accuracy and CLIPScore are reported only to verify that this spatial intervention does not conflate location following with text or global-image degradation. The clean row reuses the full-objective baseline on the four-region CVTG-2K subset in Table 3: Word Acc.=0.8309, Target IoU=PRC=0.9176, and CLIPScore=0.7925. Center error is recomputed on the same examples for the final comparison.

At a mean relative target displacement of $r = 0.35$, Target IoU decreases by only 0.0134 (0.9176 to 0.9042), while center error changes from 0.068 to 0.074. Word accuracy and CLIPScore remain effectively unchanged. These results reveal *counterfactual spatial controllability*: coordinate-span states are not merely linearly decodable spatial representations, but an executable spatial interface that the DiT follows even when the requested locations depart from the planner's self-generated layout.

E Additional Qualitative Results

Figure 8 presents additional examples beyond the benchmark prompts in the main paper. The two panels cover poster/comic and book-cover/cinematic-design settings, where the model must jointly select readable text placement and match the typography to the visual style. These results complement the controlled comparisons in Figure 3 with visually diverse applications.

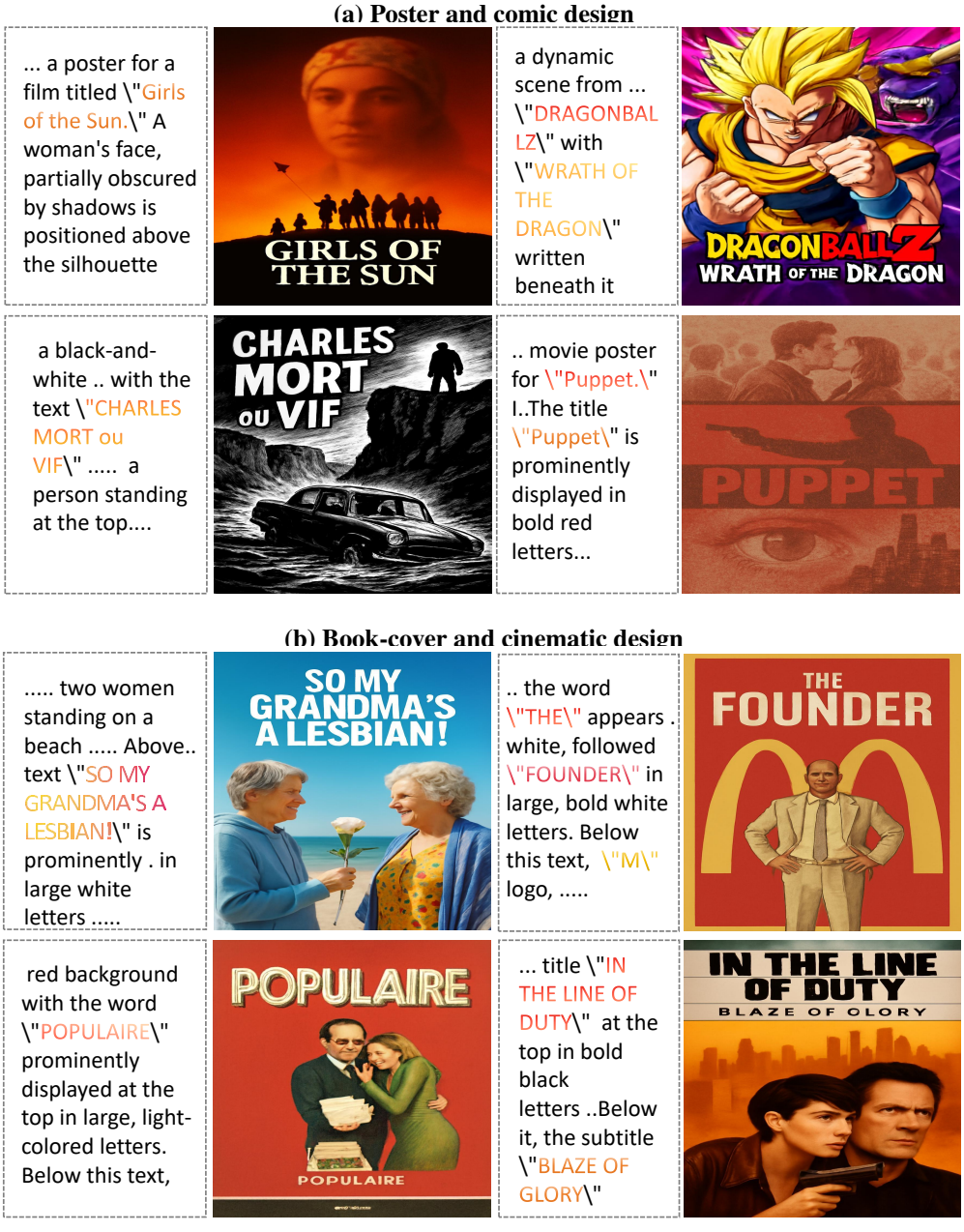


Figure 8: Additional qualitative results in diverse design settings. Panel (a) shows poster and comic examples; panel (b) shows book-cover and cinematic examples.

F Theoretical Analysis of Text-Region-Weighted Learning

Text-region weighting allocates additional supervision to the spatial support of rendered text while retaining supervision over the complete image. We analyze this design at three levels: the population

regression target, the allocation of approximation error, and the local optimization of the joint objective. The results concern the velocity-prediction objective in Eq. 4; their assumptions and proofs are given explicitly below.

Setup. Let ϑ collect the trainable parameters, and let $M(\mathbf{u})$ indicate membership in the union of the reference text boxes on the latent grid. The mask is computed from training annotations and is independent of ϑ. With $\ell_\vartheta(\mathbf{u}) = \|v_\vartheta(\mathbf{u}) - v(\mathbf{u})\|_2^2$, define

$$\begin{aligned} A(\vartheta) &= \mathbb{E}\left[\sum_{\mathbf{u}} M(\mathbf{u})\ell_\vartheta(\mathbf{u})\right], \\ B(\vartheta) &= \mathbb{E}\left[\sum_{\mathbf{u}} (1 - M(\mathbf{u}))\ell_\vartheta(\mathbf{u})\right], \\ \mathcal{L}_\gamma &= B + \gamma A, \qquad J_\gamma = R + \lambda\mathcal{L}_\gamma, \qquad R = \mathcal{L}_{\text{plan}} + \lambda_{\text{aux}}\mathcal{L}_{\text{aux}}, \end{aligned} \tag{9}$$

where $\lambda = \lambda_{\text{diff}} > 0$ and $\gamma \geq 1$. Expectations include the training examples, noise, and sampled times. Thus, J_γ is exactly the joint objective in Eq. 6. A measures error over box-covered patches, including any background within a box; it is not a glyph-segmentation or OCR loss. We assume finite second moments throughout.

F.1 Preserving the Conditional Regression Target

Proposition 1 (Consistency under layout-sufficient conditioning). Fix a conditioning representation and write $X = (\mathbf{z}_t, t, \mathbf{C}, \mathbf{u})$ and $V = v(\mathbf{u})$. Suppose $M(\mathbf{u})$ is a measurable function of X. Over unrestricted square-integrable predictors of X, the weighted and unweighted regression objectives have the same unique minimizer almost surely, $f^*(X) = \mathbb{E}[V \mid X]$. Moreover,

$$\mathcal{L}_\gamma(f) - \mathcal{L}_\gamma(f^*) = \mathbb{E}\left[\sum_{\mathbf{u}} W(\mathbf{u})\|f(X) - f^*(X)\|_2^2\right], \qquad W = 1 + (\gamma - 1)M. \tag{10}$$

Proof. Expand $\|f - V\|^2$ around f^*. The conditional cross term vanishes because $\mathbb{E}[V - f^* \mid X] = 0$ and W is X-measurable. Taking expectations and summing over patches gives Eq. 10. Since $W \geq 1$, equality with the minimum holds exactly when $f = f^*$ almost surely. □

If $\mathcal{E}_{\text{text}} = \mathbb{E}\sum_{\mathbf{u}} M\|f - f^*\|^2$ and $\mathcal{E}_{\text{bg}} = \mathbb{E}\sum_{\mathbf{u}}(1 - M)\|f - f^*\|^2$, the excess risk Δ_γ in Eq. 10 satisfies

$$\Delta_\gamma = \gamma\mathcal{E}_{\text{text}} + \mathcal{E}_{\text{bg}}, \qquad \mathcal{E}_{\text{text}} \leq \Delta_\gamma/\gamma, \qquad \mathcal{E}_{\text{text}} + \mathcal{E}_{\text{bg}} \leq \Delta_\gamma. \tag{11}$$

Consequently, a fixed excess-risk budget imposes a tighter bound on text-region approximation error while still controlling full-image velocity error. This compares error certificates at a given budget; it does not assume identical achieved budgets across training runs.

Role of the spatial interface. The assumption holds when the condition determines the reference layout, as an explicit layout does. For the learned states $\mathbf{C}$, it requires that the mask remain recoverable from those states; auxiliary coordinate supervision supports this design goal but does not by itself prove exact recoverability. For a general fixed representation, set $p_X = \mathbb{E}[M \mid X]$, $\mu_X = \mathbb{E}[V \mid X]$, and $s_X^2 = \mathbb{E}[\|V - \mu_X\|^2 \mid X]$. Conditional quadratic minimization and Cauchy–Schwarz give

$$\begin{aligned} f_\gamma^*(X) &= \frac{\mathbb{E}[WV \mid X]}{\mathbb{E}[W \mid X]}, \\ \|f_\gamma^*(X) - \mu_X\|_2 &\leq \frac{(\gamma - 1)\sqrt{p_X(1 - p_X)}\, s_X}{1 + (\gamma - 1)p_X}. \end{aligned} \tag{12}$$

Indeed, the difference equals $(\gamma-1)\mathbb{E}[(M-p_X)(V-\mu_X) \mid X]/[1+(\gamma-1)p_X]$, and the conditional variance of M is $p_X(1 - p_X)$. Thus target invariance is exact for a recoverable mask, while any shift is controlled by conditional mask uncertainty and the weight increment. This connects informative spatial conditions with reliable region weighting. The next two results allow arbitrary learned representations.

F.2 Prioritizing Text Under Shared Model Capacity

Proposition 2 (Regional priority at joint optima). Keep the parameter space, data distribution, R, and λ fixed. For $\gamma_2 > \gamma_1 \geq 1$, suppose ϑ_i globally minimizes J_{γ_i}. Then

$$A(\vartheta_2) \leq A(\vartheta_1). \tag{13}$$

More generally, if $J_{\gamma_i}(\vartheta_i) \leq \inf_\vartheta J_{\gamma_i}(\vartheta) + \delta_i$ with $\delta_i \geq 0$, then

$$A(\vartheta_2) - A(\vartheta_1) \leq \frac{\delta_1 + \delta_2}{\lambda(\gamma_2 - \gamma_1)}. \tag{14}$$

Proof. Set $F = R + \lambda B$. Approximate optimality yields $F(\vartheta_1) + \lambda\gamma_1 A(\vartheta_1) \leq F(\vartheta_2) + \lambda\gamma_1 A(\vartheta_2) + \delta_1$ and the analogous inequality with indices exchanged. Adding cancels F and gives $\lambda(\gamma_2 - \gamma_1)(A(\vartheta_2) - A(\vartheta_1)) \leq \delta_1 + \delta_2$. Setting both gaps to zero proves Eq. 13. □

This establishes regional priority even when planning and rendering share parameters and cannot fit every target perfectly. It does not require convexity, but the exact statement concerns global optima; Eq. 14 exposes the dependence on optimization quality. Background, plan, or auxiliary errors can trade off against A, explaining why a moderate weight is appropriate for a joint objective.

Effective spatial allocation and bounded weights. For a fixed grid size, let $\rho \in (0, 1)$ be the probability that a uniformly sampled patch lies inside a reference box. Under the probability measure obtained by weighting example–patch pairs by $W/\mathbb{E}[W]$, the text-region probability becomes

$$q_\gamma = \frac{\gamma\rho}{1 + (\gamma - 1)\rho}, \qquad \frac{q_\gamma}{1 - q_\gamma} = \gamma \frac{\rho}{1 - \rho}, \qquad \mathcal{L}_1 \leq \mathcal{L}_\gamma \leq \gamma\mathcal{L}_1. \tag{15}$$

The first two identities follow by normalizing the text and background masses $\gamma\rho$ and $1 - \rho$; the last follows pointwise from $1 \leq W \leq \gamma$. This is an interpretation of the existing objective, not per-image loss normalization. It shows that weighting increases the relative emphasis on text rather than uniformly rescaling the loss. The union-mask construction in Algorithm 1 keeps these bounds independent of the number of overlapping boxes, and every background patch retains a positive loss coefficient.

F.3 A Local Optimization Advantage for Text Regions

Proposition 3 (Finite-step regional descent advantage). At a common parameter point ϑ, set $a = \nabla A(\vartheta)$, $g = \nabla J_1(\vartheta)$, and $c = \lambda(\gamma - 1)$. Assume A has an L_A-Lipschitz gradient on a convex neighborhood containing the two updates $\vartheta_1^+ = \vartheta - \eta g$ and $\vartheta_\gamma^+ = \vartheta - \eta(g + ca)$, where $\eta > 0$. Then

$$A(\vartheta_\gamma^+) - A(\vartheta_1^+) \leq -\eta c\|a\|_2^2 + \frac{L_A\eta^2}{2}\left(\|g + ca\|_2^2 + \|g\|_2^2\right). \tag{16}$$

In particular, for $\gamma > 1$, $a \neq 0$, and $L_A > 0$, the weighted update has strictly smaller text-region risk whenever

$$0 < \eta < \frac{2c\|a\|_2^2}{L_A\left(\|g + ca\|_2^2 + \|g\|_2^2\right)}, \tag{17}$$

provided both updates remain in the neighborhood. *Proof.* Equation 9 gives $\nabla J_\gamma = g + ca$. Apply the smoothness upper bound to $A(\vartheta_\gamma^+)$ and the corresponding lower bound to $A(\vartheta_1^+)$. Subtracting cancels $A(\vartheta)$ and the common first-order term $-\eta\langle a, g\rangle$, leaving Eq. 16. Equation 17 makes its right-hand side negative. If $L_A = 0$, the remainder vanishes directly. □

Thus, at a fixed parameter point, region weighting contributes an additional first-order decrease $\eta\lambda(\gamma - 1)\|\nabla A\|^2$ relative to the unweighted joint update. Actual descent from $A(\vartheta)$ additionally requires $\langle a, g\rangle + c\|a\|^2 > 0$ and a sufficiently small step. This makes the mechanism explicit: the added regional term can offset a conflicting contribution from the other objectives. Proposition 3 describes full-gradient descent locally; it does not assert a global convergence-rate ordering for stochastic AdamW.

Implications for DeepFusion. Together, these results justify text-region weighting as a bounded allocation of learning emphasis: a layout-sufficient condition preserves the population target, the joint objective prioritizes text-region error under shared capacity, and the added gradient term favors local regional improvement. These properties complement the spatial interface learned with auxiliary supervision and provide an analytical basis for the region-specific execution design. The OCR and FID improvements in Table 5 provide the corresponding empirical assessment, since velocity-error guarantees alone do not imply monotonic improvements in perceptual or recognition metrics.

G LIMITATIONS

DuetGen is trained and evaluated primarily on English-centric text-image data. Preliminary qualitative evaluation on Korean and Japanese prompts suggests that its performance does not yet transfer reliably to non-Latin scripts. While the model often preserves the intended composition and approximates the visual density of the requested script, generated glyphs can exhibit inaccurate stroke structures or character identities. This observation is consistent with the limited coverage of non-Latin scripts in the current training distribution. Extending the framework with large-scale multilingual text-image data and script-aware supervision represents an important direction for future work.



Figure 9: Qualitative examples on Korean (top) and Japanese (bottom) prompts, which are underrepresented in the current training distribution. DuetGen preserves the overall composition and script-like visual appearance, but may produce inaccurate strokes or character identities.