# GlyphDraw2: Automatic Generation of Complex Glyph Posters with Diffusion Models and Large Language Models

**Jian Ma**[1], **Yonglin Deng**[2*], **Chen Chen**[1 †], **Nanyang Du**[3 *], **Haonan Lu**[1], **Zhenyu Yang**[1]
[1]OPPO AI Center
[2]The Chinese University of Hong Kong, Shenzhen
[3]Tsinghua University
majian2@oppo.com, yonglindeng@link.cuhk.edu.cn, chenchen4@oppo.com, dny22@mails.tsinghua.edu.cn,
{luhaonan,yangzhenyu}@oppo.com

## Abstract

Posters serve an essential function in marketing and advertising by improving visual communication and brand visibility, thus significantly contributing to industrial design. With the latest developments in controllable T2I diffusion models, research interest has surged in text rendering within synthesized images. Although text rendering accuracy has seen advancements, automatic poster generation remains a relatively untapped area. This paper presents an automatic poster generation framework featuring text rendering capabilities through the use of LLMs. Our framework employs a triple-cross attention mechanism based on alignment learning to achieve precise text placement within detailed contextual backgrounds. Moreover, it supports adjustable fonts, varying image resolutions, and poster rendering with textual prompts in both English and Chinese. Additionally, we present a comprehensive bilingual image-text dataset, GlyphDraw-3M, comprising 3 million image-text pairs, each with OCR annotations and resolutions exceeding 1024. Our method utilizes the SDXL architecture, and extensive experiments confirm its ability to generate posters with intricate and context-rich backgrounds.

## Introduction

Posters, as a prominent visual communication medium, have an increasing demand for personalization and customization in various fields of industrial design, whether in advertising, propaganda, marketing, or other areas. Although the powerful generative capacity of large-scale T2I diffusion models (Nichol et al. 2021; Ramesh et al. 2022; Rombach et al. 2022; Saharia et al. 2022; Podell et al. 2023) enables the creation of images with striking realism and detail, and much research effort has been devoted to addressing the limitations of text rendering in images generated by diffusion models, research on automated poster generation is still relatively limited. The goal of this paper is to endow the diffusion system with the ability to automatically generate posters. The key challenges we face include: 1) How to precisely generate fine details in small, paragraph-length text? 2) How to enhance the richness of the poster's background? 3) How to remove the need for manual user input and enable automatic poster generation based on implicit user cues?

Current advancements in visual text rendering are primarily built upon the ControlNet framework (Yang et al. 2024; Tuo et al. 2023; Zhao and Lian 2023), which generally employs glyph reference images and detailed textual layouts to guide the generation process. However, as depicted in Fig. 2, the conventional ControlNet exhibits limited control over intricate details, while the small text elements in the posters contain a significant amount of fine-grained information. Unlike the traditional global conditional control approach, glyph control is restricted to specific regions, typically covering only a small fraction of the overall image pixels, indicating a more localized control. To address these differences, we propose a triple cross-attention method. Besides the standard cross-attention computation in UNet for interacting between image latent and semantic information, we introduce two additional cross-attentions. The Q interaction information is sourced from the image's latent, while the K, V interaction information for one comes from a feature obtained from the glyph image after glyph encoding and is inserted only into the block corresponding to the SD decoder layer. This aims to capture the detailed glyph feature information and enhance the rendering precision of small text. The other K,V interaction derives from ControlNet features and aims to adaptively learn conditional information, ensuring the glyph's coherence within the overall layout. We introduced alignment target learning to enhance the richness of the poster background. Despite multiple control conditions, the aim is to align with the background output of the original semantic prompt, maintaining model integrity while ensuring accurate rendering and richness. To automate layout creation and minimize manual input, we develop comprehensive instructional datasets and fine-tune open-source LLMs for a seamless user experience during inference. Although current LLMs lack the ability to predict personalized fonts and colors, our framework inherently supports these customizations, leaving automation choices to the user.

A key requirement for poster generation is high resolution with an adjustable aspect ratio. Therefore, we use SDXL (Podell et al. 2023), which requires high-quality data. We collected specific poster data alongside open-source data to enhance SDXL training. Additionally, we use a PEA strat-

*The author did his work during internship at OPPO AI Center.
†Corresponding author.


Figure 1: GlyphDraw2 enables seamless and automated generation, eliminating the need for manual box input.

egy (Ma et al. 2023a) to make the English SDXL version multilingual in Chinese and English. Our contributions are threefold.

- We present a novel framework for the automatic generation of poster images, incorporating LLMs fine-tuning, a triple cross-attention and a semantic alignment module.
- We introduce GlyphDraw-3M, which incorporates visual text with various aspect ratios and poster content. In addition, we propose two distinct evaluation benchmarks.
- Our final results enable specific attribute controls, such as font diversity and color. Both quantitative and qualitative experimental outcomes showcase the superior performance of GlyphDraw2 in poster generation.

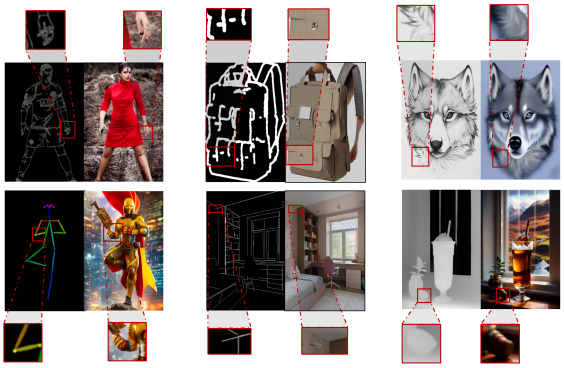

Figure 2: Details of images generated using the traditional ControlNet method. Fine-grain inconsistencies between the conditional maps and the generated images can be observed.

## Related Work

**Controllable Text-to-Image Diffusion Models.** Despite text-to-image (T2I) diffusion models have achieved astounding image generation quality, it still falls short of fulfilling all user demands. Consequently, recent studies have proposed to integrate various of conditions into T2I models to cater to more specific user needs. A notable method is model-based conditioning, wherein an auxiliary model is employed to encode additional conditioning factors, which are then incorporated into the diffusion model. For instance, IP-Adapter (Ye et al. 2023) introduces a decoupled cross-attention mechanism to separately process text and image features, significantly enhancing image-based conditioning and influencing further research (Ma et al. 2024; Wang et al. 2024a). Another significant approach is ControlNet (Zavadski, Feiden, and Rother 2023), which adds an extra encoder into the U-Net structure linked via zero convolution. This technique helps to avoid overfitting and catastrophic forgetting, allowing ControlNet to utilize specific task inputs as prior conditions for controlled generation. It has been widely researched for its applications in spatial control (Jia et al. 2024; Qin et al. 2023), text rendering (Yang et al. 2024; Zhang et al. 2023a), and 3D generation (Yu et al. 2023).

**Text Rendering.** With GlyphDraw (Ma et al. 2023b) establishing its work on text rendering last year, a series of outstanding follow-up research has appeared. We classify these developments into four categories. The first category aims to enhance the accuracy of text rendering and its background integration. GlyphDraw merges font and text attributes into

a diffusion model, whereas TextDiffuser (Chen et al. 2024) builds on this by incorporating a layout generation module and character-aware Loss. GlyphControl (Yang et al. 2024) presents ControlNet for text rendering, and AnyText (Tuo et al. 2023) adds additional conditions like text glyph, position, and masked image, along with a text perceptual loss. Brush Your Text (Zhang et al. 2023a) introduces a local attention constraint within the cross-attention layer to rectify the incorrect positioning of scene text. The second category improves character-aware text encoders. UDiffText (Zhao and Lian 2023) develops and trains a lightweight character-level text encoder to replace the standard CLIP encoder, and Glyph-ByT5 (Liu et al. 2024) further refines a character-aware ByT5 (Xue et al. 2022) encoder aligned with glyph characteristics. DreamText (Wang et al. 2024b) co-trains text encoders and generators to broadly learn and apply various fonts from the training set. SceneTextGen (Zhangli et al. 2024) utilizes a character-level encoder to draw out detailed character-specific traits. The third category mainly deals with the text layout, color, and other higher-level image attributes in the generated outputs. TextDiffuser-2 (Chen et al. 2023a) and ARTIST (Zhang et al. 2024) use LLMs to anticipate font layouts. Refining T2I Generation (Lakhanpal et al. 2024) integrates a text layout generator, and Glyph-ByT5 incorporates font type and color control during glyph-alignment pre-training. CustomText (Paliwal et al. 2024) and SceneTextGen also consider various text attribute controls. The final category enhances the base model using training data. These studies (Esser et al. 2024; Team 2024) generally yield images with high coherence, but they often show limited rendering of characters.

**LLM-Generated Text-to-Image Conditions.** Recent research (Nie et al. 2024; Zhang et al. 2023b; Gani et al. 2023) has investigated leveraging LLMs to create detailed conditions from user inputs, such as blob depictions, annotated sketches, object descriptions, and layout guidelines to direct image generation. Specifically for layouts, LayoutGPT (Feng et al. 2024) and LayoutPrompter (Lin et al. 2024) utilize LLMs to produce stylesheet languages for each object, including CSS, HTML, XML, etc. Additionally, TextDiffuser-2, LLM Blueprint (Gani et al. 2023), and Reason Out Your Layout (Chen et al. 2023b) exam the use of LLMs to generate bounding boxes (bboxes) for each object as a novel condition.

# Methodology

## Model Overview

The framework is divided into four parts, as depicted in Fig. 3. The first part, Fusion Text Encoder (FTE), mainly focuses on merging the characteristics of two modalities, namely text prompt and rendered glyph image, thereby ensuring the cohesive fusion of these modalities in the produced images. The second, and more crucial, part of our framework is the implementation of Triples of Cross-Attention (TCA). At this stage, we add two separate cross-attention layers to the SD decoder section. The first new cross-attention layer promotes the interaction between glyph features and the hidden variables within the image, thereby enhancing the precision of rendering. Concurrently, the second new cross-attention layer facilitates interaction between ControlNet features and hidden variables in the image. In the third part, we incorporate the learning of Auxiliary Alignment Loss (AAL) for semantic consistency, aiming to improve the overall layout and enrich the poster's background information. Ultimately, in the inference stage, we fine-tune LLMs to automatically interpret user descriptions and produce the corresponding glyphs and coordinate positions within the condition framework. This approach aims to achieve automatic poster generation.

## Fusion Text Encoder

This method leverages concepts from prior works such as Blip-Diffusion (Li, Li, and Hoi 2024), Subject-Diffusion (Ma et al. 2024), and AnyText, and is frequently used as a global condition control mechanism. In contrast to earlier techniques, we employed InternViT (Chen et al. 2023c), a more advanced image encoder expressly trained for character data. Initially, the input glyph condition is converted into a glyph image which is then fed into InternViT to extract the relevant features of the glyph. Mirroring the approach of AnyText, the glyph feature undergoes a linear layer for feature alignment when combined with the respective position's caption. This guarantees plug-and-play functional modularity without the need for text encoder fine-tuning.

## Triples of Cross-Attention

To guarantee precise glyph generation, we incorporate a ControlNet module as conventional. Rather than simply adding features within the decoder as previously, we propose an additional adaptive cross-attention layer positioned after the initial cross-attention layer, depicted in Fig. 3. The output of new cross-attention $S^{'}$ is computed as follows:

$$S^{'} = Attention(Q, K^{'}, V^{'}) = softmax\left(\frac{QK^{'T}}{\sqrt{d}}\right) \cdot V^{'}, \tag{1}$$

where $K^{'} = W_k^{'(j)} \cdot C^{'}$, $V^{'} = W_v^{'(j)} \cdot C^{'}$, and the $C^{'}$ features come from the corresponding block of ControlNet, $W_k^{'(j)}, W_v^{'(j)}$ are learnable projection matrices, $j$ represents the block in the U-Net decoder. Due to the asymmetric structure of SDXL's encoder and decoder layers, we ignore the interaction of the first block in the first two decoders. The goal of this method is to prevent the ControlNet of the input glyph condition, which only makes up a small portion of the generated image, from impacting the richness of the image's background. Consequently, we implement adaptive local position learning to maintain rendering accuracy while producing images with superior layouts and backgrounds.

Furthermore, the rendering accurate of paragraphs remains a significant challenge. To address this issue, we introduce a second cross-attention layer, the output of the second new cross-attention $S^{''}$ is computed as follows:

$$S^{''} = Attention(Q, K^{''}, V^{''}) = softmax\left(\frac{QK^{''T}}{\sqrt{d}}\right) \cdot V^{''}, \tag{2}$$

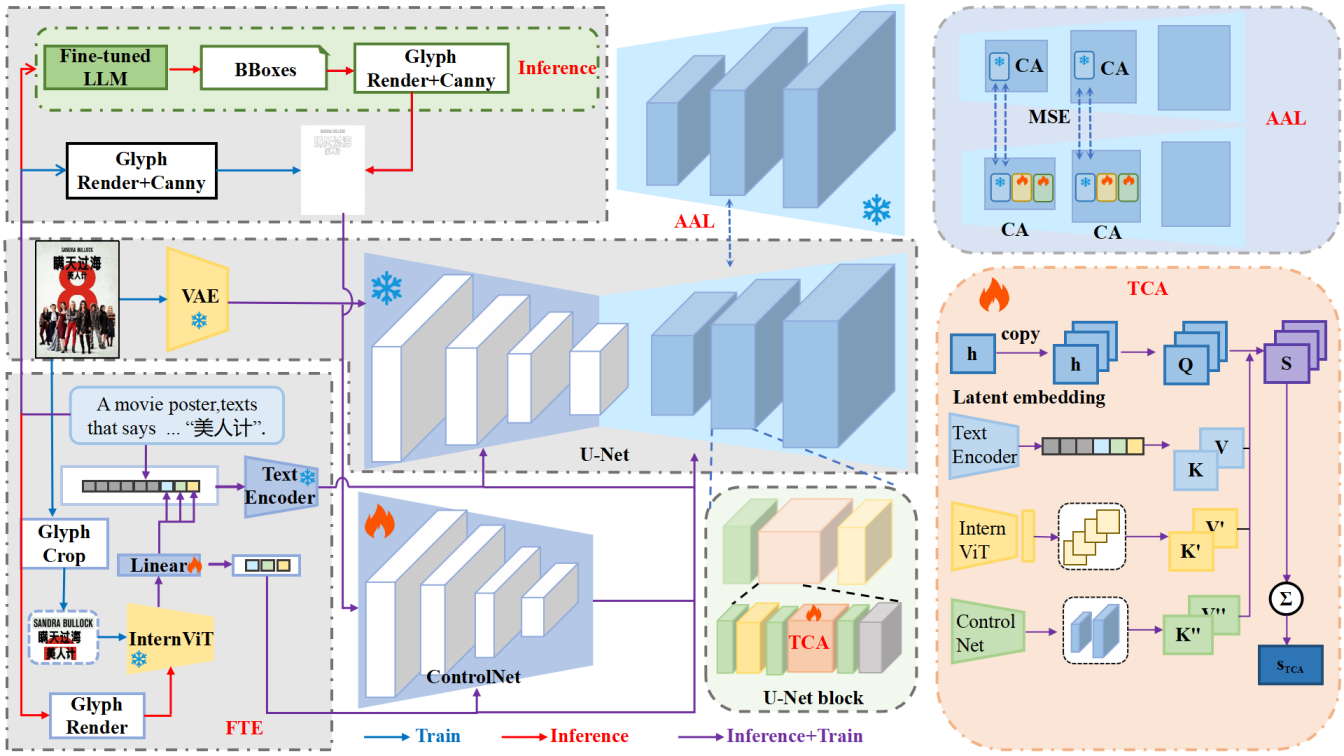


Figure 3: An overview of the proposed GlyphDraw2 framework.

where $K^{''} = W_k^{''(j)} \cdot C^{''}$, $V^{''} = W_v^{'(j)} \cdot C^{''}$, and the $C^{''}$ come from the glyph features obtained by InternViT, $W_k^{''(j)}, W_v^{''(j)}$ are learnable projection matrices, Drawing inspiration from the previous work of IP-Adapter, it is important to mention that this cross-attention layer is integrated solely within the corresponding block of the SD decoder layer. Changing the encoder layer could interfere with the features derived from the ControlNet. Numerous experiments have shown that the ControlNet's functionality relies heavily on preserving its largely unaltered encoder architecture. Additionally, it is essential for the ControlNet to retain an identical copy of the SD encoder, employing zero initialization.

In combination with the existing cross-attention layer of each block, the final TCA output is the sum of the three layers as follows:

$$S_{TCA} = \alpha S + \beta S^{'} + \gamma S^{''}, \tag{3}$$

where $\alpha, \beta, \gamma$ are constants to balance the importance of the three cross-attention layers.

## Auxiliary Align Loss

In the context of our poster generation application, besides emphasizing the accuracy of glyph generation and background harmony, we must also concentrate on the richness of the image background. Our method inevitably adds extra condition injections, which include the ControlNet feature addition and the TCA strategy, increasing the number of decoder components. The main aim of these conditions is to ensure the controllability of the generated image. However, many studies have indicated that controllability often leads to a reduction in editability or text consistency. Hence, we integrate AAL into our approach. The alignment model uses SDXL as its foundation, similarly to how ControlNet employs a duplicated SD encoder. In our method, however, we replicate the SD decoder and apply AAL between the cross-attention outputs of each block in the duplicated decoder and those in the original cross-attention layer of the TCA. The main goal of this approach is to minimize the influence of additional modules used for learning glyphs on the overall layout and image quality. Consequently, our AAL for semantic consistency $\mathcal{L}'$ can be expressed as follows:

$$\mathcal{L}' = \|softmax\left(\frac{QK^T}{\sqrt{d}}\right) \cdot V - softmax\left(\frac{QK_c^T}{\sqrt{d}}\right) \cdot V_c\|, \tag{4}$$

where $K_c$ and $V_c$ denote the CA output within each block of the replicated U-Net decoder. The ultimate loss function can be expressed as follows with a crucial hyperparameter $\lambda$:

$$\mathcal{L} = \mathbb{E}_{\mathcal{E}(x_0),C,\epsilon\sim\mathcal{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta(z_t, t, C)\|_2^2\right] + \lambda\mathcal{L}'. \tag{5}$$

## Inference with Fine-tuned LLMs

To achieve automated poster creation, the final issue that requires immediate resolution is removing the need for manual involvement, specifically in the predefined image layout process. We depend entirely on the user's caption descriptions and employ LLMs to address this challenge. Additionally,

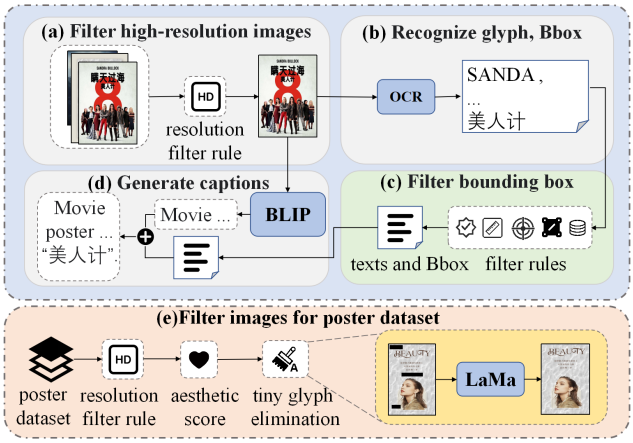


Figure 4: The procedure of training data generation.

for ease of use, we have developed our own instructional dataset and fine-tuned an open-source language model.

# Experiments

## Implementation Details and DataSet

Our model is composed of two primary components. The first is a controllable T2I poster model, with SDXL serving as the backbone of our system. To enhance the multilingual comprehension capabilities of the SDXL encoder and maintain linguistic coherence between the poster background and the generated text, we integrate the PEA-Diffusion strategy (Ma et al. 2023a) into the core structure. The second component is a layout generation model derived from LLMs. For further training details, please see the Appendix.

Previous datasets mainly focus on text-image datasets specifically designed for monolingual text rendering, such as LAION-Glyph (Yang et al. 2024) and MARIO-10M (Chen et al. 2024) for English production, which also show some limitations in terms of text layout.

Based on this, we have developed a bilingual GlyphDraw-3M. Initially, we established a data cleaning pipeline from open-source datasets as shown in Fig.4, internally collected data, and purchased data. This process includes resolution filtering, detailed PP-OCR(Li et al. 2022) recognition filtering, BLIP2 (Li et al. 2023) regeneration of image descriptions, specific aesthetic scoring filtering for poster data, and the removal of very small fonts using LaMa (Touvron et al. 2023), among other processes. In the end, we accumulated 3 million bilingual samples with detailed labels. For more dataset information, please refer to the Appendix.

## Evaluation

**Benchmark.** We aim to evaluate our method and other state-of-the-art (SOTA) techniques over five benchmarks. These consist of three publicly available datasets - **AnyText-Benchmark** (Tuo et al. 2023), **ICDAR13** (Karatzas et al. 2013), and **MARIO-Eval** (Chen et al. 2023a) - as well as two benchmarks introduced in this paper, namely **Complex-Benchmark** and **Poster-Benchmark**. More information about these benchmarks can be found in the Appendix.

**Evaluation metrics.** To evaluate these sets, we apply four metrics to determine the precision and quality of poster creation: (1) **Accuracy (Acc)**, which evaluates the ratio of accurately generated characters in the rendered text relative to the total characters required to be rendered. (2) **Normalized Edit Distance (NED)**, which is calculated using the same method as AnyText. (3) **ClipScore**, which assesses how well the generated image matches the text prompt. (4) **HPSv2**(Wu et al. 2023), which determines whether the generated images meet human preferences and serves as an indicator of image quality in terms of human preferences.

**Compared methods.** In our analysis, we evaluate a variety of methods, mainly divided into three categories. The first category consists of recently open source large-scale text generation models with rendering capabilities, such as SD3 (Esser et al. 2024), Kolors (Team 2024), and the FLUX.1 series developed by Black Forest Labs. Notably, SD3 and FLUX.1 support only English. The second category includes open-source text rendering methods, encompassing the TextDiffuser series, AnyText, UDiffText, and Glyph-ByT5. The third category comprises comparative experiments using the basic ControlNet framework.

## Experimental Results

In the following section, we present an extensive analysis of both quantitative and qualitative results, comparing our approach with SOTA methods in text rendering and poster creation. All comparative experiments are shown in Table1. GlyphDraw1.1 means that ControlNet's conditional input and InternViT's input are rendered images of static fonts. GlyphDraw2 indicates that the conditional input for ControlNet is the Canny image of the corresponding real glyph in the picture, while the InternViT input is the actual glyph from the specific picture, as illustrated in the framework in Fig.3. Furthermore, the accuracy in AnyText-Benchmark is measured using the PWAcc metric, which assesses the accuracy of words generated at specific positions, while the Acc metric is employed in other evaluation sets.

**AnyText-Benchmark.** For a fair comparison, all methods use the DDIM sampler with 50 sampling steps, a CFG scale of 9, a fixed random seed of 100 and exclude LLM. Each prompt generates a single image with the same positive and negative signals. From the results, it is clear that our model attains notably higher accuracy in rendering both Chinese and English text compared to AnyText. Furthermore, ClipScore is similar and HPSv2 is substantially better. For other approaches, TextDiffuer performs significantly worse in Chinese text generation; UDiffText does not support Chinese, and its open-source weights only support editing. Hence the metrics evaluated here result from directly editing the bbox content in AnyText-Benchmark. Additionally, it only supports generating up to 12 characters and does not support longer text generation. It is also crucial to highlight that the Glyph-ByT5 model shows certain advantages over other models, including ClipScore and HPSv2 in regards to image-text alignment and human preference metrics. However, our subjective evaluations indicated that Glyph-ByT5 occasionally fails to generate fonts within the

| Evaluation Benchmark | Model | Chinese | | | | English | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Acc | NED | ClipScore | HPSv2 | Acc | NED | ClipScore | HPSv2 |
| AnyText-Benchmark | SD3 | - | - | - | - | 0.3261 | - | 0.4517 | 0.2215 |
| | Kolors | 0.0665 | - | **0.4011** | **0.2654** | 0.0243 | - | 0.4854 | 0.2512 |
| | FLUX.1-schnell | - | - | - | - | 0.3884 | - | **0.4914** | **0.2541** |
| | ControlNet | 0.7598 | 0.8254 | 0.3749 | 0.2347 | 0.7098 | 0.8467 | 0.4558 | 0.2245 |
| | ControlNet w/ canny | 0.7804 | 0.8365 | 0.3752 | 0.2384 | 0.7954 | 0.8745 | 0.4599 | 0.2287 |
| | TextDiffuser† | 0.0605 | 0.1262 | - | - | 0.5921 | 0.7951 | - | - |
| | AnyText-v1.1 | 0.7661 | 0.8423 | 0.3968 | 0.2272 | 0.7108 | 0.8564 | 0.4721 | 0.2121 |
| | UDiffText | - | - | - | - | 0.6435 | 0.8284 | 0.4645 | 0.2214 |
| | Glyph-ByT5 | 0.7227 | 0.7799 | 0.4005 | 0.2601 | 0.7307 | 0.8353 | 0.4802 | 0.2511 |
| | **GlyphDraw1.1 w/o LLMs** | 0.7892 | 0.8476 | 0.3921 | 0.2555 | 0.7369 | 0.8921 | 0.4616 | 0.2350 |
| | **GlyphDraw2 w/o LLMs** | **0.8266** | **0.8543** | 0.3986 | 0.2589 | **0.8627** | **0.9278** | 0.4796 | 0.2451 |
| ICDAR13 | UDiffText | - | - | - | - | 0.5840 | 0.7221 | 0.4521 | 0.2101 |
| | **GlyphDraw2** | - | - | - | - | 0.6901 | 0.7629 | 0.4657 | 0.2345 |
| MARIO-Eval | TextDiffuser†† | - | - | - | - | 0.5609 | - | - | - |
| | **GlyphDraw2** | - | - | - | - | 0.7672 | 0.9330 | 0.4765 | 0.2464 |
| Complex-Benchmark | SD3 | - | - | - | - | 0.2515 | - | **0.4391** | 0.2492 |
| | Kolors | 0.0198 | - | **0.3878** | **0.2546** | 0.0033 | - | 0.4254 | **0.2546** |
| | FLUX.1-schnell | - | - | - | - | 0.2969 | - | 0.4298 | 0.2544 |
| | ControlNet | 0.6943 | 0.8745 | 0.3589 | 0.2364 | 0.2254 | 0.4025 | 0.4214 | 0.2385 |
| | ControlNet w/ canny | 0.7546 | 0.8812 | 0.3512 | 0.2386 | 0.4215 | 0.4532 | 0.4311 | 0.2298 |
| | AnyText-v1.1 | 0.5749 | 0.8560 | 0.3633 | 0.2434 | 0.0342 | 0.3755 | 0.4104 | 0.2312 |
| | Glyph-ByT5 | 0.7895 | 0.8263 | 0.3711 | 0.2455 | 0.4834 | 0.7034 | 0.4256 | 0.2412 |
| | **GlyphDraw1.1 w/o LLMs** | 0.7176 | 0.8991 | 0.3600 | 0.2422 | 0.2791 | 0.4332 | 0.4160 | 0.2395 |
| | **GlyphDraw2 w/o LLMs** | **0.9051** | **0.9037** | 0.3702 | 0.2411 | **0.5574** | **0.4928** | 0.4211 | 0.2414 |
| | LLMs+ControlNet | 0.5812 | 0.8012 | 0.3687 | 0.2365 | 0.1856 | 0.5841 | 0.4215 | 0.2356 |
| | TextDiffuser-2 | - | - | - | - | 0.0999 | 0.4428 | 0.3985 | 0.2285 |
| | LLMs+AnyText-v1.1 | 0.4850 | 0.7888 | 0.3697 | **0.2534** | 0.0455 | 0.4680 | 0.4038 | 0.2380 |
| | **GlyphDraw1.1** | 0.6215 | 0.8479 | **0.3756** | 0.2427 | 0.2264 | 0.6273 | **0.4362** | 0.2415 |
| | **GlyphDraw2** | **0.6691** | **0.7975** | 0.3754 | 0.2498 | **0.4158** | **0.6294** | 0.4312 | 0.2488 |
| Poster-Benchmark | SD3 | - | - | - | - | 0.2310 | - | 0.4128 | 0.2337 |
| | Kolors | 0.0426 | - | **0.4110** | **0.2510** | 0.0020 | - | 0.4120 | 0.2421 |
| | FLUX.1-schnell | - | - | - | - | 0.3744 | - | **0.4215** | **0.2541** |
| | ControlNet | 0.7878 | 0.8453 | 0.3844 | 0.2298 | 0.3421 | 0.7514 | 0.3902 | 0.2125 |
| | ControlNet w/ canny | 0.7911 | 0.8541 | 0.3801 | 0.2225 | 0.5012 | 0.8014 | 0.3955 | 0.2106 |
| | TextDiffuser-2 | - | - | - | - | 0.1046 | 0.3623 | 0.3914 | 0.2110 |
| | LLMs+AnyText-v1.1 | 0.7421 | 0.8894 | 0.3956 | 0.2362 | 0.2604 | 0.7120 | 0.4093 | 0.2289 |
| | Glyph-ByT5 | 0.8248 | 0.9040 | 0.4012 | 0.2366 | 0.7341 | 0.8411 | 0.4101 | 0.2354 |
| | **GlyphDraw1.1** | 0.8215 | 0.9590 | 0.3908 | 0.2378 | 0.3999 | 0.7667 | 0.3984 | 0.2297 |
| | **GlyphDraw2** | **0.8263** | **0.9585** | 0.3987 | 0.2314 | **0.7590** | **0.8759** | 0.4114 | 0.2301 |

Table 1: Evaluation Results on five benchmarks.

specified bboxes, suggesting a level of uncontrollability.

**ICDAR13.** UDiffText imposes certain limitations during the testing phase on the ICDAR13 evaluation set. For example, the authors opted to edit only 100 words for their analysis and disregarded letter casing in the evaluation process. Additionally, the Acc metric they employed is at the character level rather than the word level. We have removed these constraints and re-evaluated the ICDAR13 set using UDiffText, resulting in a comparative analysis. Our outcomes demonstrate clear superiority in four different metrics.

**MARIO-Eval.** Similarly here, the result represented by TextDiffuser†† comes from the TextDiffuser itself. Since we can't get the open-source model, we only compared the Acc metrics. Our result has a significant advantage.

**Complex-Benchmark.** In addition to comparing three large T2I models, we conduct two types of comparison experiments. The first type, based on character count and size, randomly assigns bboxes. This experiment aims to test the upper limit of the model's complex glyph generation accuracy without b-box restrictions. The second type uses fine-tuned LLMs to predict rendered characters and their corresponding b-boxes, evaluating the complex glyph generation ability in real-world scenarios. This approach provides a more in-depth evaluation and comparison of the automatic text generation functionality. Firstly, in the comparison of the three T2I models, despite Kolors' support for Chinese rendering, it is found that its ability to generate complex characters is relatively weak, with an accuracy (Acc) of only 0.02. In the English evaluation set, FLUX.1 with 12 billion parameters shows a significant advantage. Secondly, in experiments with randomly assigned bboxes, GlyphDraw2 exhibits significant advantages in both Acc and Normalized Edit Distance (NED) metrics in both Chinese and English evaluation sets. In the English evaluation set, AnyText's rendering accuracy is notably low. Although GlyphDraw2's accuracy isn't particularly high, it far exceeds AnyText's

| Model | | | | | Chinese | | |
|---|---|---|---|---|---|---|---|
| w/ CAG | w/ CAC | w/ TCA | w/ AAL | w/ FTE | Acc | ClipScore | HPSv2 |
| | | | | | 0.7782 | 0.4098 | 0.2464 |
| ✓ | | | | | 0.8014 | 0.3968 | 0.2365 |
| | ✓ | | | | 0.7845 | 0.4104 | **0.2488** |
| ✓ | ✓ | ✓ | | | 0.8154 | 0.4099 | 0.2484 |
| | | | ✓ | | 0.7689 | **0.4121** | 0.2455 |
| ✓ | ✓ | ✓ | ✓ | | 0.8122 | 0.4108 | 0.2444 |
| | | | | ✓ | 0.7841 | 0.4067 | 0.2476 |
| ✓ | ✓ | ✓ | ✓ | ✓ | **0.8161** | 0.4099 | 0.2401 |
| w/ T5 | w/ CB | w/ FT CB | w/ PP-OCR | w/ IV | Acc | ClipScore | HPSv2 |
| ✓ | | | | | 0.7951 | 0.3996 | 0.2361 |
| | ✓ | | | | 0.7981 | 0.4012 | 0.2341 |
| | | ✓ | | | 0.8017 | 0.4004 | 0.2334 |
| | | | ✓ | | 0.8014 | 0.3996 | 0.2302 |
| | | | | ✓ | **0.8263** | **0.3987** | **0.2314** |

Table 2: Ablation Results on Poster-Benchmark in Chinese.

performance. In the Chinese evaluation set, GlyphDraw2 demonstrates a slight ClipScore advantage over other methods, except Glyph-ByT5. Lastly, we compared our approach with TextDiffuser-2, which also automatically predicts bboxes. TextDiffuser-2 does not support Chinese, and its metrics in the English evaluation set are low. Analysis reveals significant issues with TextDiffuser-2's language model predictions, including incorrect and missing characters. For a fair comparison with AnyText, we use bboxes generated by our fine-tuned language models as input for AnyText. Consistent with previous results, our approach shows a substantial advantage in terms of accuracy metrics.

**Poster-Benchmark.** As shown in Table 1, performance of the three T2I large models is similar to that of the Complex-Benchmark. Beyond excellent text rendering abilities, FLUX.1 shows significant strengths in image-text alignment and human preference metrics. Moreover, it should be highlighted that Glyph-ByT5 has outperformed GlyphDraw1.1 in several metrics, and the Accuracy metric in the Chinese evaluation set is nearly equivalent to GlyphDraw2. This suggests notable advantages and potential in employing the fine-tuned ByT5 as the character encoder.

**LLMs layout prediction experiment.** We evaluate 1000 prompts at random, using the accuracy of the predicted format as our metric. Although a correct format prediction does not necessarily guarantee the correctness of the actual rendering position, such errors are typically minor. We compare three models: Qwen1.5 (Bai et al. 2023), Baichuan2 (Yang et al. 2023), and Llama2 (Touvron et al. 2023). For Qwen1.5, we test three different model sizes, whereas the other two models are evaluated with two sizes each. For additional experimental details, please see the Appendix.

## Ablation Studies

The first part uses a comparison method by integrating modules into the ControlNet base model. The second part focuses on ablation and comparison of the glyph encoder structure from InternViT as shown in Fig. 3.

**TCA.** TCA incorporates two CA layers, and each is validated separately. The w/ CAG setup, where glyph features are used as K, V for CA interaction, shows that while ACC improve, ClipScore and aesthetic metrics slightly decrease, suggesting enhanced rendering accuracy at the cost of text semantic alignment. CAC, derived from ControlNet encoder features, demonstrates that adaptive feature interaction generally boosts rendering accuracy, text semantic alignment, and aesthetic metrics. The w/ TCA setup, which involves the complete TCA module, shows improvements in ACC, and other metrics, confirming the TCA module's positive impact on rendering accuracy and image aesthetics.

**AAL.** As observed from the 5th row in the first part of Table 2, this approach enhances the semantic alignment and slightly improves image quality, though it compromises some rendering precision. Moreover, combining both the TCA and AAL strategies results in a notable improvement in metrics over using the AAL strategy alone.

**FTE.** In the seventh row of Table 2, the metrics are moderately impacted. The inclusion of glyph feature information by FTE improves rendering precision. Nevertheless, merging image modalities can reduce text semantics alignment, leading to a minor drop in ClipScore.

**Glyph encoder.** Given that the output of the glyph encoder significantly influences the entire system and feeds into the FTE module, and the CAG module within the TCA module, we run numerous experiments to show that the glyph encoder's encoding ability is positively linked to overall model performance, further proving the framework's effectiveness. The primary experiments included: 1) Encoding text directly for rendering using ByT5, which avoids the need for SentencePiece vocabulary by inputting UTF-8 bytes directly without preprocessing; 2) Encoding text directly for rendering using ChineseBERT (CB) (Sun et al. 2021), integrating glyph, pinyin, and character embeddings; 3) Expanding on experiment 2, we incorporate concepts from UDiffText and Glyph-ByT5 by fine-tuning CB within the CLIP model framework, using PP-OCR on the image side; 4) Converting text into image information first and then encoding it with PP-OCR. In this section, five experiments compare different encoders within the framework. Two main conclusions emerged. Firstly, using a text encoder directly is less effective than using a visual encoder, even with comparative learning framework fine-tuning. Secondly, a visual encoder with greater capacity and extensive training on text-related images outperforms a traditional OCR encoder.

## Conclusion

In this study, we first collected high-resolution images containing Chinese and English glyphs and subsequently constructed an automatic screening process to build a large-scale dataset. Subsequently, we establish a comprehensive framework that merges text and glyph semantics, leveraging various tiers of information to optimize rendering accuracy and richness of the background. Empirical analysis from experiments demonstrates that our methodology surpasses existing models on various evaluation sets, suggesting potential to serve as a foundation for enhancing automatic poster generation capabilities.

## References


Bai, J.; Bai, S.; Chu, Y.; Cui, Z.; Dang, K.; Deng, X.; Fan, Y.; Ge, W.; Han, Y.; Huang, F.; Hui, B.; Ji, L.; Li, M.; Lin, J.; Lin, R.; Liu, D.; Liu, G.; Lu, C.; Lu, K.; Ma, J.; Men, R.; Ren, X.; Ren, X.; Tan, C.; Tan, S.; Tu, J.; Wang, P.; Wang, S.; Wang, W.; Wu, S.; Xu, B.; Xu, J.; Yang, A.; Yang, H.; Yang, J.; Yang, S.; Yao, Y.; Yu, B.; Yuan, H.; Yuan, Z.; Zhang, J.; Zhang, X.; Zhang, Y.; Zhang, Z.; Zhou, C.; Zhou, J.; Zhou, X.; and Zhu, T. 2023. Qwen Technical Report. *arXiv preprint arXiv:2309.16609*.

Chen, J.; Huang, Y.; Lv, T.; Cui, L.; Chen, Q.; and Wei, F. 2023a. TextDiffuser-2: Unleashing the Power of Language Models for Text Rendering. *arXiv preprint arXiv:2311.16465*.

Chen, J.; Huang, Y.; Lv, T.; Cui, L.; Chen, Q.; and Wei, F. 2024. Textdiffuser: Diffusion models as text painters. *Advances in Neural Information Processing Systems*, 36.

Chen, X.; Liu, Y.; Yang, Y.; Yuan, J.; You, Q.; Liu, L.-P.; and Yang, H. 2023b. Reason out your layout: Evoking the layout master from large language models for text-to-image synthesis. *arXiv preprint arXiv:2311.17126*.

Chen, Z.; Wu, J.; Wang, W.; Su, W.; Chen, G.; Xing, S.; Zhong, M.; Zhang, Q.; Zhu, X.; Lu, L.; Li, B.; Luo, P.; Lu, T.; Qiao, Y.; and Dai, J. 2023c. InternVL: Scaling up Vision Foundation Models and Aligning for Generic Visual-Linguistic Tasks. *arXiv preprint arXiv:2312.14238*.

Esser, P.; Kulal, S.; Blattmann, A.; Entezari, R.; Müller, J.; Saini, H.; Levi, Y.; Lorenz, D.; Sauer, A.; Boesel, F.; Podell, D.; Dockhorn, T.; English, Z.; Lacey, K.; Goodwin, A.; Marek, Y.; and Rombach, R. 2024. Scaling Rectified Flow Transformers for High-Resolution Image Synthesis. arXiv:2403.03206.

Feng, W.; Zhu, W.; Fu, T.-j.; Jampani, V.; Akula, A.; He, X.; Basu, S.; Wang, X. E.; and Wang, W. Y. 2024. Layoutgpt: Compositional visual planning and generation with large language models. *Advances in Neural Information Processing Systems*, 36.

Gani, H.; Bhat, S. F.; Naseer, M.; Khan, S.; and Wonka, P. 2023. Llm blueprint: Enabling text-to-image generation with complex and detailed prompts. *arXiv preprint arXiv:2310.10640*.

Jia, C.; Luo, M.; Dang, Z.; Dai, G.; Chang, X.; Wang, M.; and Wang, J. 2024. SSMG: Spatial-Semantic Map Guided Diffusion Model for Free-form Layout-to-Image Generation. arXiv:2308.10156.

Karatzas, D.; Shafait, F.; Uchida, S.; Iwamura, M.; Bigorda, L. G. i.; Mestre, S. R.; Mas, J.; Mota, D. F.; Almazàn, J. A.; and de las Heras, L. P. 2013. ICDAR 2013 Robust Reading Competition. In *2013 12th International Conference on Document Analysis and Recognition*, 1484–1493.

Lakhanpal, S.; Chopra, S.; Jain, V.; Chadha, A.; and Luo, M. 2024. Refining Text-to-Image Generation: Towards Accurate Training-Free Glyph-Enhanced Image Generation. arXiv:2403.16422.

Li, C.; Liu, W.; Guo, R.; Yin, X.; Jiang, K.; Du, Y.; Du, Y.; Zhu, L.; Lai, B.; Hu, X.; Yu, D.; and Ma, Y. 2022. PP-OCRv3: More Attempts for the Improvement of Ultra Lightweight OCR System. arXiv:2206.03001.

Li, D.; Li, J.; and Hoi, S. 2024. Blip-diffusion: Pre-trained subject representation for controllable text-to-image generation and editing. *Advances in Neural Information Processing Systems*, 36.

Li, J.; Li, D.; Savarese, S.; and Hoi, S. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, 19730–19742. PMLR.

Lin, J.; Guo, J.; Sun, S.; Yang, Z.; Lou, J.-G.; and Zhang, D. 2024. LayoutPrompter: Awaken the Design Ability of Large Language Models. *Advances in Neural Information Processing Systems*, 36.

Liu, Z.; Liang, W.; Liang, Z.; Luo, C.; Li, J.; Huang, G.; and Yuan, Y. 2024. Glyph-ByT5: A Customized Text Encoder for Accurate Visual Text Rendering. *arXiv preprint arXiv:2403.09622*.

Ma, J.; Chen, C.; Xie, Q.; and Lu, H. 2023a. PEA-Diffusion: Parameter-Efficient Adapter with Knowledge Distillation in non-English Text-to-Image Generation. *arXiv preprint arXiv:2311.17086*.

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

Ma, J.; Zhao, M.; Chen, C.; Wang, R.; Niu, D.; Lu, H.; and Lin, X. 2023b. Glyphdraw: Learning to draw chinese characters in image synthesis models coherently. *arXiv preprint arXiv:2303.17870*.

Nichol, A.; Dhariwal, P.; Ramesh, A.; Shyam, P.; Mishkin, P.; McGrew, B.; Sutskever, I.; and Chen, M. 2021. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*.

Nie, W.; Liu, S.; Mardani, M.; Liu, C.; Eckart, B.; and Vahdat, A. 2024. Compositional Text-to-Image Generation with Dense Blob Representations. *arXiv preprint arXiv:2405.08246*.

Paliwal, S.; Jain, A.; Sharma, M.; Jamwal, V.; and Vig, L. 2024. CustomText: Customized Textual Image Generation using Diffusion Models. arXiv:2405.12531.

Podell, D.; English, Z.; Lacey, K.; Blattmann, A.; Dockhorn, T.; Müller, J.; Penna, J.; and Rombach, R. 2023. SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis. arXiv:2307.01952.

Qin, C.; Zhang, S.; Yu, N.; Feng, Y.; Yang, X.; Zhou, Y.; Wang, H.; Niebles, J. C.; Xiong, C.; Savarese, S.; et al. 2023. Unicontrol: A unified diffusion model for controllable visual generation in the wild. *arXiv preprint arXiv:2305.11147*.

Ramesh, A.; Dhariwal, P.; Nichol, A.; Chu, C.; and Chen, M. 2022. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2): 3.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695.

Saharia, C.; Chan, W.; Saxena, S.; Li, L.; Whang, J.; Denton, E. L.; Ghasemipour, K.; Gontijo Lopes, R.; Karagol Ayan, B.; Salimans, T.; et al. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35: 36479–36494.

Sun, Z.; Li, X.; Sun, X.; Meng, Y.; Ao, X.; He, Q.; Wu, F.; and Li, J. 2021. ChineseBERT: Chinese Pretraining Enhanced by Glyph and Pinyin Information. arXiv:2106.16038.

Team, K. 2024. Kolors: Effective Training of Diffusion Model for Photorealistic Text-to-Image Synthesis. *arXiv preprint*.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; Bikel, D.; Blecher, L.; Ferrer, C. C.; Chen, M.; Cucurull, G.; Esiobu, D.; Fernandes, J.; Fu, J.; Fu, W.; Fuller, B.; Gao, C.; Goswami, V.; Goyal, N.; Hartshorn, A.; Hosseini, S.; Hou, R.; Inan, H.; Kardas, M.; Kerkez, V.; Khabsa, M.; Kloumann, I.; Korenev, A.; Koura, P. S.; Lachaux, M.-A.; Lavril, T.; Lee, J.; Liskovich, D.; Lu, Y.; Mao, Y.; Martinet, X.; Mihaylov, T.; Mishra, P.; Molybog, I.; Nie, Y.; Poulton, A.; Reizenstein, J.; Rungta, R.; Saladi, K.; Schelten, A.; Silva, R.; Smith, E. M.; Subramanian, R.; Tan, X. E.; Tang, B.; Taylor, R.; Williams, A.; Kuan, J. X.; Xu, P.; Yan, Z.; Zarov, I.; Zhang, Y.; Fan, A.; Kambadur, M.; Narang, S.; Rodriguez, A.; Stojnic, R.; Edunov, S.; and Scialom, T. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. arXiv:2307.09288.

Tuo, Y.; Xiang, W.; He, J.-Y.; Geng, Y.; and Xie, X. 2023. AnyText: Multilingual Visual Text Generation And Editing. *arXiv preprint arXiv:2311.03054*.

Wang, Q.; Bai, X.; Wang, H.; Qin, Z.; and Chen, A. 2024a. Instantid: Zero-shot identity-preserving generation in seconds. *arXiv preprint arXiv:2401.07519*.

Wang, Y.; Zhang, W.; Zheng, J.; and Jin, C. 2024b. High Fidelity Scene Text Synthesis. arXiv:2405.14701.

Wu, X.; Hao, Y.; Sun, K.; Chen, Y.; Zhu, F.; Zhao, R.; and Li, H. 2023. Human Preference Score v2: A Solid Benchmark for Evaluating Human Preferences of Text-to-Image Synthesis. arXiv:2306.09341.

Xue, L.; Barua, A.; Constant, N.; Al-Rfou, R.; Narang, S.; Kale, M.; Roberts, A.; and Raffel, C. 2022. ByT5: Towards a token-free future with pre-trained byte-to-byte models. arXiv:2105.13626.

Yang, A.; Xiao, B.; Wang, B.; Zhang, B.; Bian, C.; Yin, C.; Lv, C.; Pan, D.; Wang, D.; Yan, D.; Yang, F.; Deng, F.; Wang, F.; Liu, F.; Ai, G.; Dong, G.; Zhao, H.; Xu, H.; Sun, H.; Zhang, H.; Liu, H.; Ji, J.; Xie, J.; Dai, J.; Fang, K.; Su, L.; Song, L.; Liu, L.; Ru, L.; Ma, L.; Wang, M.; Liu, M.; Lin, M.; Nie, N.; Guo, P.; Sun, R.; Zhang, T.; Li, T.; Li, T.; Cheng, W.; Chen, W.; Zeng, X.; Wang, X.; Chen, X.; Men, X.; Yu, X.; Pan, X.; Shen, Y.; Wang, Y.; Li, Y.; Jiang, Y.; Gao, Y.; Zhang, Y.; Zhou, Z.; and Wu, Z. 2023. Baichuan 2: Open Large-scale Language Models. arXiv:2309.10305.

Yang, Y.; Gui, D.; Yuan, Y.; Liang, W.; Ding, H.; Hu, H.; and Chen, K. 2024. GlyphControl: Glyph Conditional Control for Visual Text Generation. *Advances in Neural Information Processing Systems*, 36.

Ye, H.; Zhang, J.; Liu, S.; Han, X.; and Yang, W. 2023. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*.

Yu, C.; Zhou, Q.; Li, J.; Zhang, Z.; Wang, Z.; and Wang, F. 2023. Points-to-3d: Bridging the gap between sparse points and shape-controllable text-to-3d generation. In *Proceedings of the 31st ACM International Conference on Multimedia*, 6841–6850.

Zavadski, D.; Feiden, J.-F.; and Rother, C. 2023. ControlNet-XS: Designing an Efficient and Effective Architecture for Controlling Text-to-Image Diffusion Models. *arXiv preprint arXiv:2312.06573*.

Zhang, J.; Zhou, Y.; Gu, J.; Wigington, C.; Yu, T.; Chen, Y.; Sun, T.; and Zhang, R. 2024. ARTIST: Improving the Generation of Text-rich Images by Disentanglement. arXiv:2406.12044.

Zhang, L.; Chen, X.; Wang, Y.; Lu, Y.; and Qiao, Y. 2023a. Brush Your Text: Synthesize Any Scene Text on Images via Diffusion Model. arXiv:2312.12232.

Zhang, T.; Zhang, Y.; Vineet, V.; Joshi, N.; and Wang, X. 2023b. Controllable text-to-image generation with gpt-4. *arXiv preprint arXiv:2305.18583*.

Zhangli, Q.; Jiang, J.; Liu, D.; Yu, L.; Dai, X.; Ramchandani, A.; Pang, G.; Metaxas, D. N.; and Krishnan, P. 2024. Layout-Agnostic Scene Text Image Synthesis with Diffusion Models. arXiv:2406.01062.

Zhao, Y.; and Lian, Z. 2023. UDiffText: A Unified Framework for High-quality Text Synthesis in Arbitrary Images via Character-aware Diffusion Models. *arXiv preprint arXiv:2312.04884*.