

PosterMaker: Towards High-Quality Product Poster Generation with Accurate Text Rendering

Yifan Gao^{1,2,*†}, Zihang Lin^{2*}, Chuanbin Liu^{1‡}, Min Zhou²
Tiezheng Ge², Bo Zheng², Hongtao Xie¹

¹University of Science and Technology of China ²Taobao & Tmall Group of Alibaba

eafn@mail.ustc.edu.cn {liucb92, htxie}@ustc.edu.cn

{linzihang.lzh, yunqi.zm, tiezheng.gtz, bozheng}@alibaba-inc.com

Project page: <https://poster-maker.github.io>

The diagram illustrates the PosterMaker framework's end-to-end generation process, comparing it to a traditional two-stage approach.

Prompt: The box of fish oil supplements is placed on a wooden table, with a background of a serene ocean and clear sky, symbolizing purity and the natural source of the product.

Subject: A box of ABC Deep-Sea Fish Oil Omega-3.

Text: abc深海鱼油 迷你颗粒 支持心脏

Poster: An image of the fish oil box with overlaid text: abc deep-sea fish oil support the heart.

Previous: two stage: This section shows a sequential process:

- User Input: The fish oil box.
- Sence Generation: The box is shown on a wooden table with a beach background.
- Text Rendering: The final image with overlaid text.

Ours: end to end: This section shows a direct, single-step generation process:

- User Input: The fish oil box.
- PosterMaker: The process step.
- End2End Generation: The final image with overlaid text.

(a): Examples of generated posters for various products:

- Skincare and Beauty Lady's Milk: 护肤美颜贵妇乳 99.9%纯度玻色因
- Nestlé BEBA supreme Baby Milk Powder: 增强免疫力 幼儿奶粉 易于冲调
- Double-colored Concealer: 双色的眉粉 持久不脱色 防水防汗
- Simple Wooden Bookshelf: 简约的实木书架 多种颜色 防尘功能
- PureNest Essential Oil: 清新花香 白色铃兰
- Natural Mineral朱砂: 天然原矿朱砂 原创设计
- Manicure Practice Base: 美甲练习底座 多色款式 快速上手
- Yiping Canning: 一品制罐 密封保鲜 多色选择

(b): Examples of generated posters for various products:

- ABC Deep-Sea Fish Oil: abc深海鱼油 迷你颗粒 支持心脏
- Skincare and Beauty Lady's Milk: 护肤美颜贵妇乳 99.9%纯度玻色因
- Double-colored Concealer: 双色的眉粉 持久不脱色 防水防汗
- Simple Wooden Bookshelf: 简约的实木书架 多种颜色 防尘功能
- PureNest Essential Oil: 清新花香 白色铃兰
- Natural Mineral朱砂: 天然原矿朱砂 原创设计
- Manicure Practice Base: 美甲练习底座 多色款式 快速上手
- Yiping Canning: 一品制罐 密封保鲜 多色选择

Figure 1. (a) Definition of the advertising product poster generation task. The input includes the prompt, subject image, and the texts to be rendered with their layouts. The output is the poster image. (b) The comparison of our method with the previous method. PosterMaker generates posters end-to-end, while previous methods first generate poster backgrounds and then render texts. (c) Visualization results demonstrate that PosterMaker can generate harmonious and aesthetically pleasing posters with accurate texts and maintain subject fidelity.

Abstract

Product posters, which integrate subject, scene, and text, are crucial promotional tools for attracting customers. Cre-

ating such posters using modern image generation methods is valuable, while the main challenge lies in accurately rendering text, especially for complex writing systems like Chinese, which contains over 10,000 individual characters. In this work, we identify the key to precise text rendering

* Equal contribution. ‡ Corresponding author.

[†] Work done during the internship at Alibaba Group.

as constructing a character-discriminative visual feature as a control signal. Based on this insight, we propose a robust character-wise representation as control and we develop TextRenderNet, which achieves a high text rendering accuracy of over 90%. Another challenge in poster generation is maintaining the fidelity of user-specific products. We address this by introducing SceneGenNet, an inpainting-based model, and propose subject fidelity feedback learning to further enhance fidelity. Based on TextRenderNet and SceneGenNet, we present PosterMaker, an end-to-end generation framework. To optimize PosterMaker efficiently, we implement a two-stage training strategy that decouples text rendering and background generation learning. Experimental results show that PosterMaker outperforms existing baselines by a remarkable margin, which demonstrates its effectiveness.

1. Introduction

Product posters, which showcase items for sale within well-chosen background scenes and include descriptive text, play a vital role in e-commerce advertising by capturing customers' attention and boosting sales. Creating such posters necessitates photographing the product in carefully selected environments that highlight its features, as well as thoughtfully choosing text colors and fonts to ensure that the text is appealing, legible, and harmonious with the background. This process can be quite expensive. With the significant advancements in large-scale text-to-image (T2I) models [13, 35, 39], synthesizing such product posters with image generation models attracts increasing attention. In this paper, we focus on the product poster generation task. Specifically, given a prompt describing the background scene, the foreground image of the user-specified subject and some texts together with their layouts, we aim to develop a model to generate the subject into the desired scene background and accurately render the text in an end-to-end manner (as shown in Fig. 1 (a)).

A straightforward solution for this task is to first generate the subject into the desired scene [2, 11, 40], and then predict the text attributes (such as color and font) [14, 23] and render them on the image. However, such two-stage approach suffers from disharmony between the text and the poster background (as shown in Fig. 2 (b)). And collecting training data is also challenging since the text attributes, especially the text font, are difficult to extract from the poster. Another solution is learning to generate the poster using a per-pixel synthesis approach, which can benefit from directly learning the distribution of professionally designed posters. We focus on such one-stage solution. The main challenge is how to ensure the text rendering accuracy.

Many recent works [13, 25, 42, 49] have been proposed to improve the text rendering accuracy for large diffusion models. Great progress has been made and some

recent work can achieve high rendering accuracy for English. However, for non-Latin languages like Chinese, one of the most widely spoken languages, achieving high rendering accuracy remains challenging. This difficulty stems from the existence of over 10,000 characters, with Chinese characters characterized by complex and diverse stroke patterns, making it extremely difficult to train a model to memorize the rendering of each individual character. Recent studies [4, 28, 42] have focused on extracting visual features of text as control signals. Typically, these approaches render text lines into glyph images and extract **line-level** text **visual** features to guide generation.

Nevertheless, line-level visual features often lack the discriminative power to capture character-level visual nuances. To address this limitation, GlyphByT5 [25, 26] introduced a box-level contrastive loss with sophisticated glyph augmentation strategies to enhance character-level discriminativeness, achieving promising results. In this paper, we point out that the key to high-accuracy text rendering lies in constructing **character-discriminative visual features** as control signals. Specifically, we render each character as a glyph image and extract visual features via a visual encoder. These features are then concatenated with positional embeddings to form a character-level representation. Then we propose TextRenderNet, an SD3 [13] controlnet-like [53] architecture that takes the character-level representation as the control signal to render visual text. Our experiments demonstrate that the proposed character-level representation is effectively capable of achieving accurate text rendering.

In the task of poster generation, another important thing is to generate the user-specific subject into a desired scene while keeping high subject fidelity. Recent subject-driven controllable generation methods [40, 44, 51] can synthesize such images, but they still cannot ensure that the user-specified subject is completely consistent in the generated details (e.g., the logo on the product may be inaccurately generated), which could potentially mislead customers. Therefore, we follow poster generation methods [5, 11, 22] to address this task via introducing an inpainting-based module named SceneGenNet. However, we found that even using inpainting methods, subject consistency is not always achieved as the inpainting model sometimes extends the subject shape (as shown in Fig. 2 (a)). Similar phenomenon is also observed in [11, 12]. To address this issue, we elaboratively develop a detector to detect the foreground extension cases. Then we employ the detector as a reward model to train the SceneGenNet via feedback learning for further improving subject fidelity.

Combining the proposed TextRenderNet and SceneGenNet, we develop a framework named PosterMaker that can synthesize the product poster in an end-to-end manner. To efficiently optimize PosterMaker, we introduce a two-stage



Figure 2. The illustration of the three challenges faced by poster generation, which seriously hinder the practical application.

training strategy to separately train TextRenderNet and SceneGenNet. This training strategy decouples the learning of text rendering and background image generation, thus TextRenderNet and SceneGenNet can focus on their specific tasks. Qualitative results (as shown in Fig. 1 (c)) demonstrate our training strategy is effective for training Poster-Maker and it achieves promising poster generation results.

To summarize, our contributions are as follows:

- We proposed a novel framework named PosterMaker, which mainly consists of a TextRenderNet and a SceneGenNet. With a two-stage training strategy, PosterMaker can synthesis aesthetically product posters with texts accurately and harmoniously rendered on it.
- We reveal the core of achieving accurate Chinese text rendering is to construct a robust character-level text representation as the control condition. These findings can inspire future research on improving the text rendering abilities of T2I models.
- We improve the subject fidelity via subject fidelity feedback learning, which is shown effective in addressing the subject inconsistency issue.

2. Related Work

2.1. Poster Generation

Generating posters involves combining various elements like a subject image, a background scene image, and text to ensure the subject and text are prominently and accurately displayed while maintaining an appealing look. Automating this process is quite complex and challenging. Methods like AutoPoster [23], Prompt2Poster [45], and COLE [16] break it down into stages: creating images and layout, predicting the visual properties of text, and rendering the poster. These approaches have several steps and often struggle to precisely obtain all the necessary visual attributes like font and color gradients. With the emergence of more advanced generative models [35], methods like JoyType [19], Glyph-byt5 [25], and GlyphDraw2 [28] can directly generate the image and text together at the pixel level based on the poster prompt, text content, and layout. This more streamlined approach can leverage more readily available poster pixel data for training, but there is still room for improvement in terms of the overall poster cohesion and text accuracy. Our method is also a one-stage, direct pixel-level generation approach that simultaneously creates the image and

text. However, our focus is on generating posters for a given product subject, where the input includes the subject image, prompt, text content, and layout. In addition to considering text rendering accuracy and overall poster harmony, we also need to maintain the fidelity of the product.

2.2. Visual Text Rendering

Recently, text-to-image (T2I) models [1, 13, 41] have made significant strides in enhancing English text rendering by introducing stronger text encoders, such as T5 [38]. However, multilingual text image generation still faces significant challenges due to the large number of non-Latin characters and complex stroke structures. Early work [49] has explored the ControlNet-based method [53], using low-level visual images such as glyph images as the control signal for text image generation. However, glyph images are easily affected by text size and shape, especially complex stroke details. Besides, some recent works [4, 27, 28, 42, 52, 55] utilize more robust visual features, such as line-level OCR features as control conditions to further improve the text accuracy. But the line-level visual features still perform poorly in representing stroke details for each character. To address this issue, GlyphByT5 [25, 26] proposes a method with box-level contrastive learning to align the text features extracted from the language model with the features extracted from the visual encoder. To effectively learn such alignment, GlyphByT5 relies on collecting massive amounts of data and developing complex data augmentation strategies for the alignment pre-training, which lacks flexibility. In contrast, in this paper, we reveal that the key to high-accuracy text rendering lies in constructing discriminative character-level visual features. Thus we propose a plug-and-play and robust character-level text representation derived from off-the-shelf OCR encoders, which can accurately represent the visual structure of the text without additional training and enable precise text rendering.

2.3. Subject-Preserved Scene Generation

To create a scene image with a product subject while ensuring subject fidelity, two main methods are commonly used. One is the subject-driven method [3, 6, 20, 36, 40], which adjusts the position, angle and lighting of the subject based on the prompt to create a harmonious image. However, it often struggles to preserve the significant features of the subject. The other utilizes inpainting-based background com-

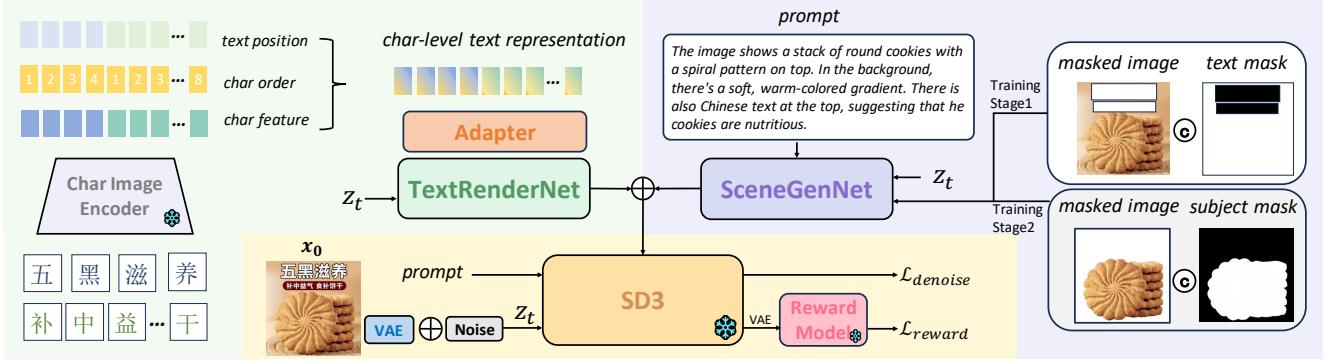


Figure 3. The framework of the PosterMaker, which is based on the SD3. To precisely generate multilingual texts and create aesthetically pleasing poster scenes, TextRenderNet and SceneGenNet are introduced, whose outputs are used as control conditions added to the SD3.

pletion techniques [2, 11, 43]. It only generates the non-subject areas of an image and naturally keeps consistency in the original subject area. But it sometimes extends the foreground subject [11, 12], such as adding an extra handle to a cup, which also reduces subject fidelity. To maximize subject fidelity, our method uses background completion and a reward model to determine whether the foreground extension occurred, thereby enhancing subject fidelity.

3. Method

3.1. Problem Formulation

This paper focuses on the creation of product posters, which typically consist of multiple elements such as text, subjects, and scenes, as illustrated in Fig. 1 (a). The central challenge of this task is to generate these elements accurately and harmoniously, offering both research and practical applications. The task is defined as:

$$I_g = f(I_s, M_s, T, P), \quad (1)$$

where I_g denotes the generated poster image, I_s represents the subject image, and M_s is the subject mask. The variable T signifies the content and the position of text and P is the prompt describing the background scene. Subsequent sections will detail the design of PosterMaker, and our proposed solution to this task.

3.2. Framework

As shown in Fig. 3, PosterMaker is developed based on Stable Diffusion 3 (SD3) [13], which contains a strong VAE for reconstructing the image details like text stroke. And we propose two modules, i.e., TextRenderNet and SceneGenNet, to address the poster generation task. TextRenderNet is specifically designed to learn visual text rendering, taking character-level visual text representations as input to achieve precise and controllable text rendering. SceneGenNet, on the other hand, accepts a masked image (indicating which content should remain unchanged) and a prompt, learning to generate the foreground subject within the desired scene described by the prompt. Both TextRenderNet

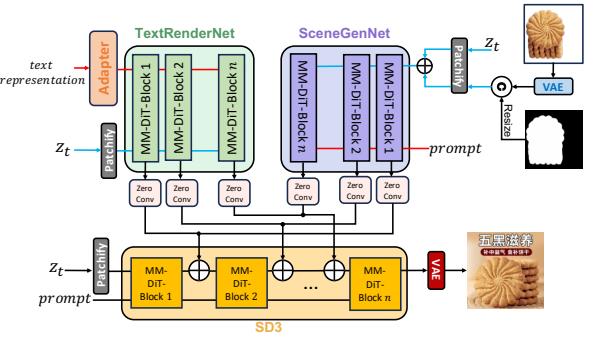


Figure 4. The details of TextRenderNet and SceneGenNet, showcasing their model architectures and their interactions with SD3.

and SceneGenNet are grounded in a ControlNet-like [52] architecture derived from SD3 and their architectures are detailed in Fig. 4. They share the same internal structure, comprising several cascaded MM-DiT blocks [13], with weights copied from the base model for initialization. The output of each MM-DiT block is added to the corresponding block of the base model after passing through a zero convolution layer [53]. The key distinction between the two modules lies in their input configurations. SceneGenNet takes the prompt as input to the text condition branch, and for the visual branch, the input is derived by the latent feature at timestep t , the subject mask, and the masked latent to preserve the foreground area. In contrast, TextRenderNet receives text representations (detailed in the next section) in the text condition branch for text rendering. An adapter, consisting of a linear layer and layer normalization, adjusts the feature dimensions of these text representations before they are input to TextRenderNet. The outputs of each block in TextRenderNet and SceneGenNet are directly added to the corresponding block outputs of the SD3 base model.

3.3. Character-level Visual Representation for Precise Text Rendering

Recently, some works have explored multilingual visual text generation. Among them, a promising approach is based on ControlNet-like methods [42], which utilize both glyph images and line-level OCR features as conditions.



Figure 5. The distinction between the multilingual character-level text representation we proposed and the line-level methods of previous works like AnyText [42] and GlyphDraw2 [28].

However, this control information cannot accurately represent characters: 1) glyph images are easily affected by text size and shape, making them less robust. 2) line-level visual features lack fine-grained stroke features and are limited by the OCR model’s poor capability to recognize long texts. To address these challenges, this paper proposes a plug-and-play and robust character-level text representation, where each character is precisely represented by one token.

Specifically, the text C has n characters. For each character c_i , its feature is separately extracted by a pre-trained OCR encoder f_v and then averaged and pooled to obtain a compact character representation vector $r_{c_i} \in \mathbb{R}^c$. Thus, the character-level text representation is defined as follows:

$$r_{c_i} = \text{avgpool}(f_v(I_{c_i})), \quad (2)$$

$$R_c = [r_{c_1}, r_{c_2}, \dots, r_{c_n}], \quad (3)$$

where I_{c_i} is the i -th character image rendered in a fixed font, and $R_c \in \mathbb{R}^{n \times c}$ is the char-level text representation.

As shown in Fig. 5, compared to previous methods, our key difference is extracting representations from character glyph images. This enables the model to perceive character stroke structures and achieve high text accuracy. Additionally, since the number of characters is fixed, we can pre-extract the representations of each character and store them in a dictionary, eliminating the need for online rendering and feature extraction. This significantly simplifies the training and inference pipeline.

Finally, this text representation lacks order and positional information. Thus, the character order encoding P_{rank} is introduced to represent the order of characters in the text, which is implemented through a sinusoidal position encoding of the char order. Besides, inspired by GLIGEN [21], the text position coordinates are mapped to sinusoidal position encoding P_{bbox} to control the position of the text. Then we concatenate P_{rank} , P_{bbox} and R_c along the feature dimension to construct the final text representation.

3.4. Improving Subject Fidelity

In the task of generating product posters, it is crucial to maintain subject fidelity, i.e., ensuring that the subject in the generated poster remains consistent with the user-specified subject. To achieve this goal, we employ SceneGenNet to perform background inpainting, which is trained to precisely preserve the foreground subject and only inpaint the background according to the prompt. However, inpainting-based models sometimes extend the foreground subject into

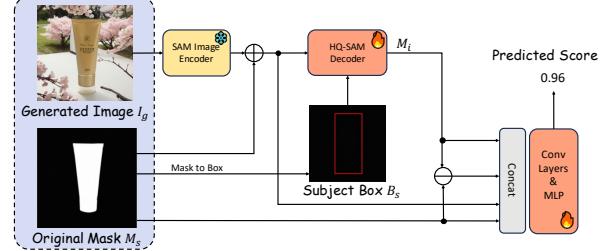


Figure 6. The model details of the foreground extension detector.

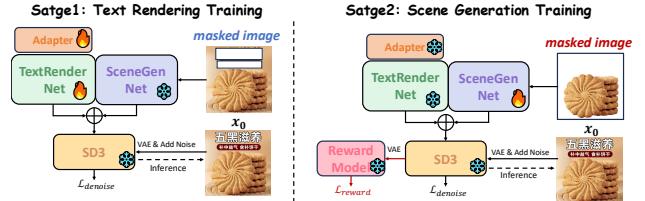


Figure 7. The illustration of our two-stage training strategy for efficiently optimizing PosterMaker.

another subject (as shown in Fig. 2 (b)) thereby compromising subject fidelity. We refer to this as “foreground extension”. To mitigate this issue, we develop a model to detect foreground extension and employ it as a reward model to fine-tune PosterMaker to improve subject fidelity.

Foreground Extension Detector. We develop the foreground extension detector S_θ based on HQ-SAM [17]. As shown in Fig. 6, we input the generated image I_g to SAM [18] image encoder. The subject mask M_s and box B_s are provided as mask prompt and box prompt, respectively, to the HQ-SAM decoder to obtain an intermediate mask M_i . Next, we concatenate the image features extracted from SAM encoder with M_s , M_i and $M_s - M_i$ at the channel dimension. The concatenated features are processed through convolutional layers and MLP layers to predict whether the foreground has been extended in the generated image. We collected 20k manually annotated images to train the foreground extension detector S_θ .

Subject Fidelity Feedback Learning. The foreground extension detector S_θ , after the offline training, is used as a reward model to supervise PosterMaker to improve subject fidelity. Specifically, assuming the reverse process has a total of T' steps, we follow ReFL [47] to first sample $z_{T'} \sim \mathcal{N}(0, 1)$ and after $T' - t'$ steps of inference ($z_{T'} \rightarrow z_{T'-1} \rightarrow \dots \rightarrow z_{t'}$), we obtain $z_{t'}$, where $t' \sim [1, t_1]$. Then, we directly perform a one-step inference $z_{t'} \rightarrow z_0$ to accelerate the reverse process. Furthermore, z_0 is decoded to the generated image x_0 . The detector S_θ predicts the foreground extension score for x_0 , and this score is used as the reward loss to optimize the generator G_ϕ (i.e., PosterMaker). The reward loss is defined as follows:

$$\begin{aligned} \mathcal{L}_{\text{reward}}(\phi) &= -\mathbb{E}_{(x, c, m) \sim \mathcal{D}_{\text{train}}, t' \sim [1, t_1], z_{T'} \sim \mathcal{N}(0, 1)} \\ &\log \sigma(1 - S_\theta(G_\phi(z_{T'}, x, c, m, t'), m)), \end{aligned} \quad (4)$$



Figure 8. Qualitative comparison with different methods. Best viewed on Screen. To aid comprehension, Chinese text lines in the image are translated into English and annotated using corresponding colors.

where x, c, m sampled from the train data $\mathcal{D}_{\text{train}}$, represent the subject image, control conditions, and subject mask respectively. To avoid overfitting, we don't calculate reward loss for the cases where the foreground extension score is below 0.3. Our total training loss is defined as:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{denoise}} + \lambda \mathcal{L}_{\text{reward}}, \quad (5)$$

where λ is the hyperparameter to adjust the weight of reward loss and the denoise loss.

3.5. Training Strategy

To efficiently train PosterMaker, this paper introduces a two-stage training strategy, as shown in Fig. 7, aimed at decoupling the learning for text rendering and background image generation. Specifically, in the first stage, the training task is local text editing. We freeze SceneGenNet and only the TextRenderNet and adapter are optimized. Since we initialize SceneGenNet with pre-trained weights of inpainting-controlnet [7], it can fill the local background well thus TextRenderNet can focus on learning text generation. In the second stage, the training task is subject-based text-to-image generation. Here we froze TextRenderNet and only train the SceneGenNet. In this stage, SceneGenNet focuses on learning poster scenes and creative design from the train data. Notably, Stage 1 learns local text editing/inpainting and Stage 2 learns background inpainting, thus the input images indicating the area to inpaint are different (See Fig. 7). With such a two-stage training strategy, TextRenderNet and SceneGenNet can be efficiently optimized since they can focus on their specific tasks.

4. Experiments

4.1. Experimental Setup

Dataset. We crawl product posters from online e-commerce platforms to construct our training set. Our training data mainly consists of Chinese posters, we first employ PPOCRv4 model [34] to extract the text content and their bounding boxes from the images as a coarse annotation. And we ask some annotators to further refine the bounding boxes and correct the text content to improve the annotation quality. Resulting in a dataset containing 160k images. We generate image captions with GPT4-o [32] and extract foreground subject masks with U²-Net [37] and VitMatte [50]. We randomly select 302 images for evaluation and leave the rest for training. To better evaluate the performance of our method, we use LLM [10] to generate some background prompts and text layouts as evaluation samples, after manually checking and removing those irrational ones, we obtain another 198 evaluation samples to form a final evaluation set named PosterBenchmark containing 500 samples.

Evaluation Metrics. We follow Anytext [42] to evaluate text rendering accuracy using two metrics: sentence accuracy (Sen. Acc) and normalized edit distance (NED). Specifically, we crop the text line from the generated image according to the provided bounding box and utilize the OCR model [31] to predict the content s_{pred} of the generated text line. We denote the ground truth text content as s_{gt} . A text line is considered to be correctly generated if $s_{\text{pred}} = s_{\text{gt}}$; this condition is used to calculate Sen. Acc. Additionally, we compute the normalized edit distance (NED) between s_{pred} and s_{gt} to measure their similarity. We further calculate FID [15] to measure the visual quality and CLIP-T [40] metric for evaluating text-image alignment.



Figure 9. Qualitative comparison using various text features. It is obvious that the character-level OCR features we used (PPOCR Char) are the most effective at maintaining character accuracy.

Implementation Details. Our SceneGenNet is initialized from pre-trained SD3 Inpainting-Controlnet [7] and TextRenderNet is initialized from SD3 [13] weight with the same configuration as in [8]. For Subject Fidelity Feedback Learning, we follow existing work [47] to uniformly sample t' between [1, 10]. Within this range, the one-step inference result of image x_0 from t' is close to the full inference result. The weight coefficient of λ is set to 0.0005. The learning rate is set to 1e-4 and the batch size is set to 192. We train our framework for 26k and 29.5k steps for training stage1 and stage2, respectively. Finally, PosterMaker was trained on 32 A100 GPUs for 3 days. During the sampling process, based on the statistical information, a maximum of 7 lines of text and 16 characters per line of text are selected from each image to render onto the image, as this setting can cover most situations in the dataset.

4.2. Comparison with Prior Works

Baseline methods. We carefully designed the following baseline approaches¹ based on existing open-sourced techniques for comparative analysis. **SD3_inpaint_byt5:** We encode the text content into prompt embeddings using ByT5 [48] and employ an adapter to map these embeddings to the original prompt embedding space of SD3 before feeding them into the controlnet, which enables the controlnet to render multilingual text. **SD3_canny&inpaint:** First render the text into a white-background image and extract the canny edge from it as control. Then finetune a pre-trained SD3 canny controlnet together with an inpainting controlnet to achieve multilingual text rendering. **Anytext:** It is the SOTA open-sourced T2I method that supports multilin-

Model	Sen.	ACC ↑	NED ↑	FID ↓	CLIP-T ↑	FG Ext. Ratio ↓
SD3.inpaint.AnyText	52.78%	75.27%	100.87	26.90	14.82%	
SD3.inpaint.byt5	52.28%	86.57%	65.45	26.71	14.60%	
AnyText	63.90%	82.81%	71.27	26.69	19.25%	
Glyph-ByT5-v2	69.54%	87.65%	79.23	26.60	18.91%	
SD3.canny&inpaint	80.75%	92.75%	67.19	27.03	14.38%	
GlyphDraw2	86.14%	96.78%	72.49	26.72	16.52%	
GT (w/ SD1.5 Rec.)	76.95%	89.91%	-	-	-	
GT (w/ SD3 Rec.)	98.09%	99.36%	-	-	-	
GT	98.53%	99.59%	-	-	-	
Ours (SD1.5)	72.12%	88.01%	68.17	26.93	-	
Ours	93.36%	98.39%	65.35	27.04	11.57%	

Table 1. Comparison with baseline methods.

gual text rendering and its text editing mode supports text inpainting [42]. So we directly finetune it on our data using its text editing training pipeline. **SD3_inpaint_Anytext:** First generate the background with SD3 inpainting controlnet, then render the text on the corresponding region using Anytext. **Glyph-ByT5-v2** and **GlyphDraw2:** They are both the SOTA T2I methods that support multilingual text rendering [26, 28]. However, they don't have open-sourced pre-trained weights, so we reproduced them on our dataset. And we added an inpainting controlnet for them to support subject-preserved generation.

Quantitative Comparison. We trained all baseline models on the same dataset, and then quantitatively compared all methods on the PosterBenchmark, as shown in Tab. 1. It is worth noting that SD3 is used as the base model by default, but since we observed that the SD1.5 VAE leads to significant error in reconstruction, to enable a more equitable comparison between our method and AnyText (SD1.5 architecture), we also implemented an SD1.5 version of PosterMaker with the same experimental setup as AnyText. As the VAEs, especially SD1.5, introduce some reconstruction error and the OCR model may incorrectly recognize some characters, we also report the metrics on ground truth

¹Details can be found in the Appendix.

Text Feature	Type	Sen. ACC	NED
ByT5	textual feat.	33.48%	54.50%
Canny	img	81.50%	92.72%
TrOCR Line	visual feat.	26.58%	49.46%
TrOCR Char	visual feat.	94.27%	98.54%
PPOCR Line	visual feat.	38.91%	53.86%
PPOCR Char (Ours)	visual feat.	95.15%	98.75%
GT (w/o Rec.)	-	98.53%	99.59%
GT (w/ SD3 Rec.)	-	98.09%	99.36%

Table 2. Quantitative comparison using various text features.

Method	FG Ext. Ratio↓	Sen. ACC ↑	NED↑	FID↓	CLIP-T↑
Ours	11.57%	93.36%	98.39%	65.35	27.04
Ours w/o \mathcal{L}_{reward}	15.05%	93.11%	98.21%	65.10	27.04

Table 3. Evaluation on the subject fidelity feedback learning.

images as an upper bound. As shown in Tab. 1, our method achieves the best performance on all metrics. Notably, on text rendering metrics Sen. ACC and NED, our model outperforms the baselines by an impressive margin and is already close to the upper bound. The promising results demonstrate the effectiveness of the proposed PosterMaker.

Qualitative Comparison. The results are shown in Fig. 8. Compared to the baselines, our PosterMaker generates more readable and accurate poster images with texts, particularly for smaller texts. Notably, as an end-to-end generation method, PosterMaker automatically creates underlays to enhance the contrast between text and background, effectively highlighting the text. This feature is crucial in product poster design for capturing viewers’ attention. These findings demonstrate that our PosterMaker successfully learns the distribution of posters created by human designers.

4.3. Ablation Study and Analysis

How to achieve high text rendering accuracy? We conduct experiments to explore the effectiveness of different control conditions for visual text rendering. Due to the fact that text rendering accuracy is primarily determined by the first training stage, we discard the second training stage in this experiment to save computational resources. The results are summarized in Tab. 2. We observed several valuable experimental results: 1) The use of char-level features significantly outperforms previous line-level features, benefiting from finer-grained representation. This explains why previous methods [4, 28, 42], achieve inferior performance (PPOCR Line is used in [28, 42], TrOCR Line is used in [4]). Recent concurrent works [29, 46] have also found similar experimental findings as ours. 2) Char-level feature representation is superior to low-level image features such as Canny. 3) PPOCR outperforms TrOCR, which is attributed to PPOCR being a multi-language OCR model, while TrOCR is an English version model. 4) Even though TrOCR has not been trained on multi-language text data, it still achieves decent results, likely because it extracts universal visual structural features. 5) ByT5 extracts char-level features but the performance is inferior to OCR features, because it extracts semantic features rather than character structural features, while T2I models’ text rendering



Figure 10. Visual examples showing the effect of \mathcal{L}_{reward} .

capability relies more on character structural features. We present visualization results in Fig. 9. We observe that when using line-level features as a control, the generated text occasionally becomes completely unrecognizable. This suggests that line-level features are insufficient for achieving precise text rendering. Additionally, it is evident that using canny control always introduces stroke artifacts, particularly in smaller texts (as seen in row 3 of Fig. 9). This further demonstrates that canny control is also not an ideal condition for text rendering. In summary, the char-level feature extracted by PPOCR performs optimally and the accuracy is already close to the upper bound, indicating *the discriminative char-level visual feature is the key to achieve high text rendering accuracy*.

Effectiveness of subject fidelity feedback learning. We calculate the foreground extension ratio (termed as FG Ext. Ratio) by asking human annotators to manually check each generated image whether the foreground subject is incorrectly extended. As demonstrated in Tab. 3, training our model with \mathcal{L}_{reward} effectively reduces FG Ext. Ratio by 3.4%, while maintaining subtle variations in other performance metrics. Representative visual examples are presented in Fig. 10. Besides, our model outperforms baseline methods in FG Ext. Ratio (see Tab. 1). These results show the efficacy of our proposed subject fidelity feedback learning approach in mitigating foreground extension artifacts.

5. Conclusion

The application of image generation in poster creation is often impeded by subpar text rendering and inconsistent subjects. To address these challenges, this paper introduces a novel framework, PosterMaker, which synthesizes aesthetically pleasing product posters with accurate and harmonious texts and contents. Moreover, we reveal that the key underlying successful multilingual text rendering is the construction of robust character-level visual text representations. Additionally, we propose subject fidelity feedback learning to mitigate inconsistencies in subjects. Through extensive experiments, our method demonstrates a significant improvement in both high-precision text generation and subject fidelity. These findings not only advance poster generation but also inspire future research on T2I models.

Acknowledgments

This work was supported by the National Nature Science Foundation of China (62425114, 62121002, U23B2028, 62232006, 62272436) and Alibaba Group (Alibaba Research Intern Program).

References

- [1] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Qinsheng Zhang, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, et al. ediff-i: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022. 3
- [2] Tingfeng Cao, Junsheng Kong, Xue Zhao, Wenqing Yao, Junwei Ding, Jinhui Zhu, and Jiandong Zhang. Product2img: Prompt-free e-commerce product background generation with diffusion model and self-improved LMM. In *Proceedings of the 32nd ACM International Conference on Multimedia, MM 2024, Melbourne, VIC, Australia, 28 October 2024 - 1 November 2024*, pages 10774–10783. ACM, 2024. 2, 4
- [3] Kelvin C. K. Chan, Yang Zhao, Xuhui Jia, Ming-Hsuan Yang, and Huisheng Wang. Improving subject-driven image synthesis with subject-agnostic guidance. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 6733–6742. IEEE, 2024. 3
- [4] Haoxing Chen, Zhuoer Xu, Zhangxuan Gu, Jun Lan, Xing Zheng, Yaohui Li, Changhua Meng, Huijia Zhu, and Weiqiang Wang. Diffute: Universal text editing diffusion model. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. 2, 3, 8
- [5] Ruidong Chen, Lanjun Wang, Weizhi Nie, Yongdong Zhang, and An-An Liu. Anyscene: Customized image synthesis with composited foreground. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8724–8733, 2024. 2
- [6] Wenhui Chen, Hexiang Hu, Yandong Li, Nataniel Ruiz, Xuhui Jia, Ming-Wei Chang, and William W. Cohen. Subject-driven text-to-image generation via apprenticeship learning. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. 3
- [7] Alimama Creative. Sd3-controlnet-inpainting. <https://huggingface.co/alimama-creative/SD3-Controlnet-Inpainting>, 2024. 6, 7, 2, 4
- [8] Alimama Creative. Sd3-controlnet-softedge. <https://huggingface.co/alimama-creative/SD3-Controlnet-Softedge>, 2024. 7, 2
- [9] Alimama Creative. Ecomxl-controlnet-inpaint. https://huggingface.co/alimama-creative/EcomXL_controlnet_inpaint, 2024. 2
- [10] Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Bin Wang, Linke Ouyang, Xilin Wei, Songyang Zhang, Haodong Duan, Maosong Cao, Wenwei Zhang, Yining Li, Hang Yan, Yang Gao, Xinyue Zhang, Wei Li, Jingwen Li, Kai Chen, Conghui He, Xingcheng Zhang, Yu Qiao, Dahua Lin, and Jiaqi Wang. Internlm-xcomposer2: Mastering free-form text-image composition and comprehension in vision-language large model. *arXiv preprint arXiv:2401.16420*, 2024. 6, 1
- [11] Zhenbang Du, Wei Feng, Haohan Wang, Yaoyu Li, Jingsen Wang, Jian Li, Zheng Zhang, Jingjing Lv, Xin Zhu, Junsheng Jin, et al. Towards reliable advertising image generation using human feedback. In *European Conference on Computer Vision*, pages 399–415. Springer, 2024. 2, 4, 3
- [12] Amir Erfan Eshratifar, Joao V.B. Soares, Kapil Thadani, Shaunak Mishra, Mikhail Kuznetsov, Yueh-Ning Ku, and Paloma De Juan. Salient object-aware background generation using text-guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 7489–7499, 2024. 2, 4
- [13] 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. 2, 3, 4, 7, 1
- [14] Yifan Gao, Jinpeng Lin, Min Zhou, Chuanbin Liu, Hongtao Xie, Tiezheng Ge, and Yuning Jiang. Textpainter: Multi-modal text image generation with visual-harmony and text-comprehension for poster design. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 7236–7246, 2023. 2
- [15] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6626–6637, 2017. 6
- [16] Peidong Jia, Chenxuan Li, Yuhui Yuan, Zeyu Liu, Yichao Shen, Bohan Chen, Xingru Chen, Yinglin Zheng, Dong Chen, Ji Li, Xiaodong Xie, Shanghang Zhang, and Baining Guo. Cole: A hierarchical generation framework for multi-layered and editable graphic design, 2024. 3
- [17] Lei Ke, Mingqiao Ye, Martin Danelljan, Yifan Liu, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu. Segment anything in high quality. In *Advances in Neural Information Processing Systems*, pages 29914–29934. Curran Associates, Inc., 2023. 5
- [18] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4015–4026, 2023. 5, 3
- [19] 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. 3
- [20] Dongxu Li, Junnan Li, and Steven C. H. Hoi. Blip-diffusion: Pre-trained subject representation for controllable text-to-

- image generation and editing. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. 3
- [21] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22511–22521, 2023. 5
- [22] Zhaochen Li, Fengheng Li, Wei Feng, Honghe Zhu, An Liu, Yaoyu Li, Zheng Zhang, Jingjing Lv, Xin Zhu, Junjie Shen, et al. Planning and rendering: Towards end-to-end product poster generation. *arXiv preprint arXiv:2312.08822*, 2023. 2
- [23] Jinpeng Lin, Min Zhou, Ye Ma, Yifan Gao, Chenxi Fei, Yangjian Chen, Zhang Yu, and Tiezheng Ge. Autoposter: A highly automatic and content-aware design system for advertising poster generation. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 1250–1260, 2023. 2, 3
- [24] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022. 1
- [25] 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*, pages 361–377. Springer, 2024. 2, 3
- [26] Zeyu Liu, Weicong Liang, Yiming Zhao, Bohan Chen, Ji Li, and Yuhui Yuan. Glyph-byt5-v2: A strong aesthetic baseline for accurate multilingual visual text rendering. *arXiv preprint arXiv:2406.10208*, 2024. 2, 3, 7
- [27] Zhiying Lu, Chuanbin Liu, Xiaojun Chang, Yongdong Zhang, and Hongtao Xie. Dhvt: Dynamic hybrid vision transformer for small dataset recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025. 3
- [28] Jian Ma, Yonglin Deng, Chen Chen, Haonan Lu, and Zhenyu Yang. Glyphdraw2: Automatic generation of complex glyph posters with diffusion models and large language models. *arXiv preprint arXiv:2407.02252*, 2024. 2, 3, 5, 7, 8
- [29] Lichen Ma, Tiezhu Yue, Pei Fu, Yujie Zhong, Kai Zhou, Xiaoming Wei, and Jie Hu. Chargen: High accurate character-level visual text generation model with multimodal encoder. *arXiv preprint arXiv:2412.17225*, 2024. 8
- [30] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jianjun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2022. 4
- [31] ModelScope. https://modelscope.cn/models/damo/cv_convnexttiny_ocr-recognition-general_damo/summary, 2023. 6
- [32] OpenAI. <https://openai.com/index/hello-gpt-4o/>, 2024. 6
- [33] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al.
- Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 3
- [34] PaddlePaddle. <https://github.com/PaddlePaddle/PaddleOCR>, 2023. 6, 2, 3
- [35] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: improving latent diffusion models for high-resolution image synthesis. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. 2, 3
- [36] Tianhao Qi, Shancheng Fang, Yanze Wu, Hongtao Xie, Jiawei Liu, Lang Chen, Qian He, and Yongdong Zhang. Deadiff: An efficient stylization diffusion model with disentangled representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8693–8702, 2024. 3
- [37] Xuebin Qin, Zichen Zhang, Chenyang Huang, Masood Dehghan, Osmar Zaiane, and Martin Jagersand. U2-net: Going deeper with nested u-structure for salient object detection. page 107404, 2020. 6
- [38] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and PeterJ. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv: Learning, arXiv: Learning*, 2019. 3
- [39] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 10674–10685. IEEE, 2022. 2
- [40] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22500–22510, 2023. 2, 3, 6
- [41] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022. 3
- [42] Yuxiang Tuo, Wangmeng Xiang, Jun-Yan He, Yifeng Geng, and Xuansong Xie. Anytext: Multilingual visual text generation and editing. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. 2, 3, 4, 5, 6, 7, 8
- [43] Haohan Wang, Wei Feng, Yaoyu Li, Zheng Zhang, Jingjing Lv, Junjie Shen, Zhangang Lin, and Jingping Shao. Generate e-commerce product background by integrating category commonality and personalized style. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2025. 4
- [44] Qixun Wang, Xu Bai, Haofan Wang, Zekui Qin, and Anthony Chen. Instantid: Zero-shot identity-preserving gener-

- ation in seconds. *arXiv preprint arXiv:2401.07519*, 2024.
- [45] Shaodong Wang, Yunyang Ge, Liuhan Chen, Haiyang Zhou, Qian Wang, Xinhua Cheng, and Li Yuan. Prompt2poster: Automatically artistic chinese poster creation from prompt only. In *Proceedings of the 32nd ACM International Conference on Multimedia, MM 2024, Melbourne, VIC, Australia, 28 October 2024 - 1 November 2024*, pages 10716–10724. ACM, 2024.
- [46] Tong Wang, Xiaochao Qu, and Ting Liu. Textmastero: Mastering high-quality scene text editing in diverse languages and styles. *arXiv preprint arXiv:2408.10623*, 2024.
- [47] Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imageward: Learning and evaluating human preferences for text-to-image generation. In *Advances in Neural Information Processing Systems*, pages 15903–15935. Curran Associates, Inc., 2023.
- [48] Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. ByT5: Towards a token-free future with pre-trained byte-to-byte models. *Transactions of the Association for Computational Linguistics*, 10:291–306, 2022.
- [49] Yukang Yang, Dongnan Gui, Yuhui Yuan, Weicong Liang, Haisong Ding, Han Hu, and Kai Chen. Glyphcontrol: Glyph conditional control for visual text generation. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.
- [50] Jingfeng Yao, Xinggang Wang, Shusheng Yang, and Baoyuan Wang. Vitmatte: Boosting image matting with pre-trained plain vision transformers. *Information Fusion*, 103:102091, 2024.
- [51] Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*, 2023.
- [52] Boqiang Zhang, Zuan Gao, Yadong Qu, and Hongtao Xie. How control information influences multilingual text image generation and editing? *arXiv preprint arXiv:2407.11502*, 2024.
- [53] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023.
- [54] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Computer Vision and Pattern Recognition*, 2016.
- [55] Yuanzhi Zhu, Jiawei Liu, Feiyu Gao, Wenyu Liu, Xinggang Wang, Peng Wang, Fei Huang, Cong Yao, and Zhibo Yang. Visual text generation in the wild. In *European Conference on Computer Vision*, pages 89–106. Springer, 2024.

PosterMaker: Towards High-Quality Product Poster Generation with Accurate Text Rendering

Supplementary Material

Due to space limitations, we were unable to present all experimental results in the main text. In this supplementary material, we will give more details about our experiments and present additional results.

6. Implementation Details

Training and Inference. We fully follow the settings of SD3 [13]. During training, the denoise loss $\mathcal{L}_{\text{denoise}}$ uses simplified flow matching, also known as 0-rectified flow matching loss [24]. In inference, we also use the inference method of flow matching, with 28 inference steps.

TextRenderNet and SceneGenNet. TextRenderNet and SceneGenNet have an architecture similar to SD3 [13], composed of multiple MM-DiT Blocks. In our implementation, TextRenderNet consists of 12 layers of MM-DiT Blocks, while SceneGenNet consists of 23 layers of MM-DiT Blocks. The output of the N_i -th block of SceneGenNet is first added with the output of the $\lceil \frac{N_i}{2} \rceil$ -th block of TextRenderNet, and then add to the N_i -th SD3 block.

Classifier-Free Guidance. We use CFG during inference, with a CFG scale of 5. Additionally, since the “prompt” inputted to TextRenderNet is not a caption but a text representation, the negative one for CFG is set to a zero vector. During training, we randomly drop the text representation to a zero vector with 10% probability.

The Setting of t_1 in Reward Loss. We follow [47] to train the reward loss at the last 10 inference steps, i.e., we set t_1 to 10. Within the range of $t' \sim [1, t_1]$, the result of the image x_0 obtained by one-step inference is close to the result of complete inference.

Details about Metric Calculation. Our evaluation benchmark contains samples generated by LLM [10] thus there is no ground truth for these samples. Therefore, we exclude these LLM-generated samples when calculating metrics that depend on ground truth images, i.e., FID metric for all experiments, text accuracy metrics for GT (with and without VAE reconstruction) and results for ablation on different text features.

About ground truth for training Foreground Extension Detector. We treat the task of detecting foreground extension as a binary classification problem and ask annotators to manually label the ground truth.

7. Baseline Details

We carefully designed 6 baseline approaches based on existing techniques for comparative analysis. The de-

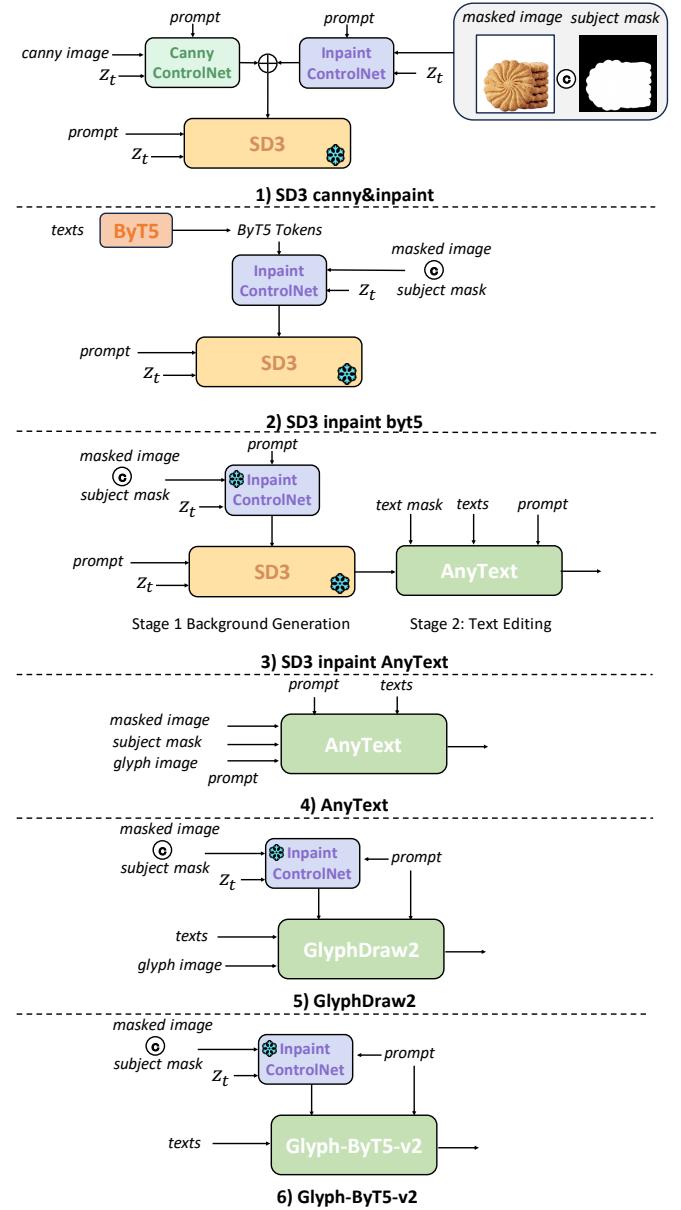


Figure 11. Detailed illustration of the implementation of the different baseline methods.

tails are shown in Fig. 11. For 1) SD3_inpaint_byt5, 2) SD3_canny&inpaint, and 4) AnyText, we fine-tune them on our 160K dataset for the poster generation task. Meanwhile, 3) SD3_inpaint_Anytext is a two-stage inference method. In the first stage, the pre-trained Inpaint ControlNet gener-

ates the background, and in the second stage, AnyText performs the text editing task, with AnyText also fine-tuned on the 160K dataset specifically for the text editing task. The Inpainting ControlNet is initialized from pre-trained SD3 Inpainting-Controlnet [7] and Canny ControlNet is initialized from [8]. For 5) GlyphDraw2 [28] and 6) Glyph-ByT5-v2 [26] are both the SOTA T2I methods that support multilingual text rendering. However, they neither have open-source pre-trained weights nor support subject input, so we reproduced them on our dataset by adding the pre-trained inpainting controlnet [9] to support the subject input.

8. Scalable Training for Text Rendering

Our proposed two-stage training strategy allows the model to learn two different capabilities (i.e., text rendering and scene generation) separately, enabling more flexibility with distinct datasets for each phase. Recent text rendering methods [4, 25, 26, 42] typically train their models on datasets containing millions of samples. To verify the potential of further improving our performance with more training data, we build a large dataset with 1 million samples and we directly obtain the text annotations with PPOCRv4 [34] without manually annotating. And we use this dataset for the first stage of text rendering training and use the same 160k data for the second stage of scene generation learning. Compared to using 160k data in both of the previous stages, the text sentence accuracy significantly improved by 4.48% (as shown in Tab. 4), demonstrating that the multi-stage training strategy is flexible and scalable. However, in the main experiments, we select to report the performance of our model training only on 160k data for fair comparison with the baselines.

Data Size (St.1 & St.2)	Sen. ACC	NED
160k & 160k	93.11%	98.21%
1M & 160k	97.59%	99.38%

Table 4. Quantitative comparison with different data sizes for text rendering training.

9. Discussion on advantages of end-to-end over two-stage methods.

The main weakness of two-stage methods (first inpaint background, then render text) is their inability to consistently provide a clean background for texts (see Fig. 12, reducing text readability, especially with complex backgrounds. In contrast, one-stage methods generate texts and backgrounds simultaneously, enabling them to create a clean backdrop or overlays that enhance text visibility.

10. Text Position Control

The position control of PosterMaker uses a very straightforward approach (as shown in Fig. 13), mapping the text

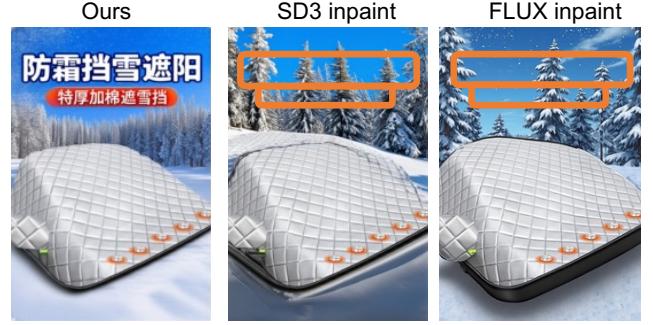


Figure 12. Showcases for end-to-end and two-stage methods.

Method	mIoU	IoU@0.5	IoU@0.7
Ours	84.65%	97.18%	93.94%

Table 5. Evaluation on text location accuracy.

bounding box to cosine position encoding, which is then concatenated with text features and used as the input to TextRenderNet. To demonstrate our method’s effectiveness, we evaluate the bounding box IoU (Intersection of Union) metric as follows: 1) we employ OCR model to extract texts from the generated image. 2) For each ground truth text, we identify the best-matched OCR-detected text based on edit distance and then calculate the IoU between their corresponding bounding boxes. We average the IoU score over all the samples to obtain mean IoU (termed mIoU). And we also report IoU@R which indicates the proportion of samples with IoU higher than R . As shown in Tab. 5, our method achieves a high mIoU of 84.65% and 93.94% samples have an IoU score higher than 0.7. These promising results prove that our text position control method is simple yet effective.

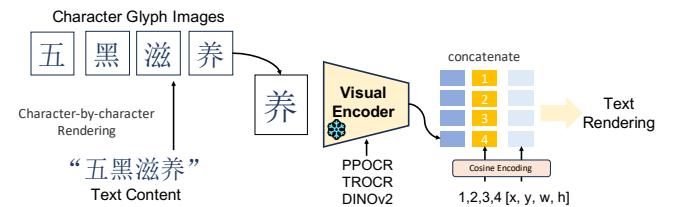


Figure 13. Detailed illustration of how we construct the position embedding for controlling the text position.

11. Comparison Between GlyphByT5 and PosterMaker

GlyphByT5 [25, 26] are recently proposed visual text rendering methods that achieve high text rendering accuracy. And we will discuss some differences and internal connections between our PosterMaker and GlyphByT5 on how to control text rendering.

- Text position control: GlyphByT5 achieve text position control by modifying the original cross-attention module with their proposed region-wise multi-head cross-attention. In contrast, our PosterMaker encodes the text location directly into the character-level text representation to accomplish text position control. As discussed in Sec. 10, our approach is both simple and effective for precise text location control.
- Text content control: both GlyphByT5 and our Poster-Maker control the generation of text content by constructing suitable text representation. Specifically, in this work, we claim that the key to achieve accurate text rendering is to extract **character-level visual** features as the control condition and carefully construct a robust text representation based on off-the-shelf OCR model [34]. In GlyphByT5, the authors also extract **character-level text** features, but with a *textual* encoder named ByT5 [48]. Then they propose glyph-alignment pre-training to align these *textual* features with pre-trained *visual* encoders DINOv2 [33]. Additionally, they employ box-level contrastive learning with complex augmentations and a hard-mining strategy to enhance *character-level* discriminativeness. We hypothesize that the primary reason both our method and GlyphByT5 achieve high text rendering accuracy is our shared goal of constructing a robust **character-level visual** representation. In fact, the ability of GlyphByT5’s character-level visual representation is distilled from the pre-trained *visual* encoder DINOv2, rather than inherited from the pre-trained *textual* encoder ByT5 itself. In order to verify our hypothesis and insights, we adopt a more direct approach to directly replace the PPOCR encoder in PosterMaker with DINOv2. As shown in Tab. 6, simply extracting character-wise visual features with DINOv2 can also achieve precise text rendering. This result further verifies our claim: the key to precise text rendering is to extract **character-level visual** features as the control condition.

Text Feature	Type	Sen. ACC	NED
PPOCR Line	visual feat.	38.91%	53.86%
PPOCR Char	visual feat.	95.15%	98.75%
DINOv2 Line	visual feat.	4.25%	20.59%
DINOv2 Char	visual feat.	94.92%	98.66%
GT (w/o Rec.)	-	98.53%	99.59%
GT (w/ SD3 Rec.)	-	98.09%	99.36%

Table 6. Quantitative comparison using various text features.

12. Visualization of Training Samples

We present example training images from our dataset in Fig. 17. The dataset predominantly consists of Chinese text, with a small portion of English text. Additionally, it in-

cludes challenging cases with small-sized text elements.

13. The Generalization of Text Representation.

PosterMaker is trained primarily on common Chinese data, with only a minimal amount of English data. Despite this, it demonstrates a notable level of generalization, enabling it to generate English, Japanese, and uncommon Chinese characters that were not included in the training set (as shown in Fig. 16). In order to quantitatively evaluate the generalization capability of PosterMaker, we compared the accuracy of different text representations on uncommon characters using a randomly sampled uncommon character benchmark. The results show that our method can also generalize well to some characters that are unseen in the training set. Our performance is inferior to the canny baseline, likely because the canny baseline has been pre-trained on large-scale image data.

Text Feature	Type	Sen. ACC	NED
ByT5	textual feat.	2.01%	10.27%
Canny	img	65.12%	74.56%
PPOCR Line	visual feat.	8.34 %	15.84%
PPOCR Char	visual feat.	61.54%	70.38%

Table 7. Quantitative comparison of the rendering results of different text features on uncommon characters.

14. Ablation about Foreground Extension Detector

We collected 20k manually annotated images to train the foreground extension detector. We randomly selected 10% samples as a validation set, while using the remaining 90% for model training. We conduct ablation experiments on different architecture designs of the detector to verify the effectiveness of the proposed architecture. We implement 2 baselines: 1) **RFNet** [11]: we reimplemented RFNet based on the description in their paper [11]. Since we could not access their depth and saliency detection models, we modified our implementation to only use the product image and generated image as input, excluding the depth and saliency maps. 2) **RFNet(SAM)** : in this baseline, we replace the image encoder used in RFNet with the same SAM[18] im-

Method	Precision	Recall	F1 Score
RFNet (our impl.)	76.52%	75.52%	76.02%
RFNet (SAM)	81.35%	80.99%	81.17%
Ours	83.52%	84.81%	84.16%

Table 8. Evaluation on different architectures of foreground extension detector.

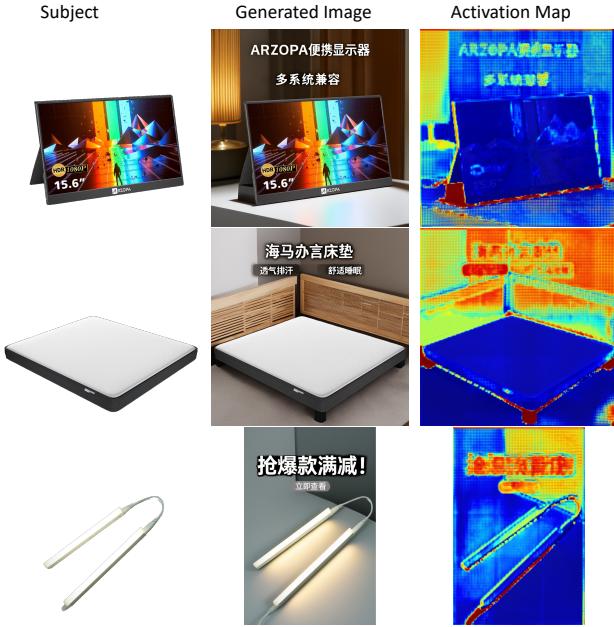


Figure 14. Class activation map of the foreground extension detector.

age encoder used in our method. As summarized in Tab. 8, our proposed foreground extension detector outperforms the baselines by a considerable margin, which demonstrates its effectiveness.

In Fig. 14, we visualize the class activation map [54] of our proposed foreground extension detector. As shown, we can observe a notably higher activation score in the extended foreground regions compared to other areas. This compelling evidence demonstrates that our detector has effectively learned to discern foreground extension cases, thereby it can serve as a robust reward model for fine-tuning PosterMaker to mitigate the foreground extension problem.

15. Ablation about SceneGenNet

SceneGenNet enables our model to perform background inpainting while preserve the subject so we cannot directly remove it. We replace it by SDEdit [30] to achieve inpainting. As the results shown in Sec. 15, replacing it results in a significant drop of performance.

Model	Sen. ACC \uparrow	NED \uparrow	FID \downarrow	CLIP-T \uparrow
Ours w/o SceneGenNet	90.53%	97.95%	79.44	26.67
Ours	93.36%	98.39%	65.35	27.04

Table 9. Comparison between SceneGenNet and SDEdit

16. Discussion on the impact of the test set size.

To ensure a fairer comparison between PosterMaker and the baseline methods, we expanded the test set to 5,000 sam-

ples(10x the previous PosterBenchmark). The results are shown in Tab. 10, and the experimental conclusions remain consistent with the previous test set. Due to the calculation principle of the FID metric, increasing the test size leads to a significant decrease in the FID scores for all methods, but still maintains the same conclusion.

Model	Sen. ACC \uparrow	NED \uparrow	FID \downarrow	CLIP-T \uparrow
Glyph-ByT5-v2	67.87%	86.23%	20.37	21.08
SD3_canny&inpaint	74.49%	88.78%	17.91	20.79
GlyphDraw2	83.81%	96.49%	15.24	20.67
Ours	90.20%	97.58%	13.36	21.36

Table 10. Comparison with baseline methods on 5,000 test samples.

17. Discussion on the meaningless texts generated outside target position.

In our early experimental attempts about text rendering in poster generation, we found that the trained model sometimes generates meaningless texts outside the target area of the text, which will seriously affect the aesthetics. We conjecture that the main reason is that the ground truth images sometimes contain text outside the specified position. To solve this problem, we masked out the extra text during training and it solved most cases.

Specifically, SceneGenNet is initialized from pre-trained SD3 Inpainting-Controlnet [7]. In the second stage of training, we simultaneously mask out the regions of the untrained texts (usually those that are too small or just logos) both in the subject mask input to SceneGenNet and in the ground truth image used for loss calculation(as shown in Fig. 15). It is worth noting that although these small texts and logos are not included in the training, we have also annotated them to address the aforementioned issues. Finally, this technique makes the loss corresponding to the masked-out regions very close to zero so that the model will not learn these meaningless texts.



Figure 15. Example of our solution technique for meaningless texts and logos that generated outside target position.



Figure 16. Visualization results on texts in English, Japanese, and uncommon Chinese characters.



Figure 17. Visualization of ground truth for some samples in the dataset.