Rethinking Layered Graphic Design Generation with a Top-Down Approach

Jingye Chen[1,2], Zhaowen Wang[2], Nanxuan Zhao[2], Li Zhang[2], Difan Liu[2], Jimei Yang[3], Qifeng Chen[1]
[1]HKUST [2]Adobe Research [3]Runway
qwerty.chen@connect.ust.hk zhawang@adobe.com cqf@ust.hk

Abstract

Graphic design is crucial for conveying ideas and messages. Designers usually organize their work into objects, backgrounds, and vectorized text layers to simplify editing. However, this workflow demands considerable expertise. With the rise of GenAI methods, an endless supply of high-quality graphic designs in pixel format has become more accessible, though these designs often lack editability. Despite this, non-layered designs still inspire human designers, influencing their choices in layouts and text styles, ultimately guiding the creation of layered designs. Motivated by this observation, we propose ***Accordion****, a graphic design generation framework taking the* ***first attempt*** *to convert AI-generated designs into editable layered designs, meanwhile refining nonsensical AI-generated text with meaningful alternatives guided by user prompts. It is built around a vision language model (VLM) playing distinct roles in three curated stages: (1) reference creation, (2) design planning, and (3) layer generation. For each stage, we design prompts to guide the VLM in executing different tasks. Distinct from existing bottom-up methods (e.g., COLE and Open-COLE) that gradually generate elements to create layered designs, our approach works in a top-down manner by using the visually harmonious reference image as global guidance to decompose each layer. Additionally, it leverages multiple vision experts such as SAM and element removal models to facilitate the creation of graphic layers. We train our method using the in-house graphic design dataset Design39K, augmented with AI-generated design images coupled with refined ground truth created by a customized inpainting model. Experimental results and user studies by designers show that Accordion generates favorable results on the DesignIntention benchmark, including tasks such as text-to-template, adding text to background, and text de-rendering, and also excels in creating design variations.*

1. Introduction

Graphic design is an important media format for modern visual communication. Specifically, graphic design is intrinsically constructed in a layered format, including distinct layers for foreground objects, background, and vectorized text. This structured layering allows for editability and customization. Despite the significant utility, creating layered graphic designs is a prohibitive task for most people due to the need for design expertise and huge effort. With the aid of image GenAI models, more design images have become available in rasterized pixel format. While they are visually compelling, they inherently lack editability. Even for simple operations such as horizontal flipping, text becomes unreadable since they are not separated from the background or other elements. Although users can employ some image editing tools [5, 14, 15, 23] to modify the attributes of elements, such an approach is inconvenient compared to op-

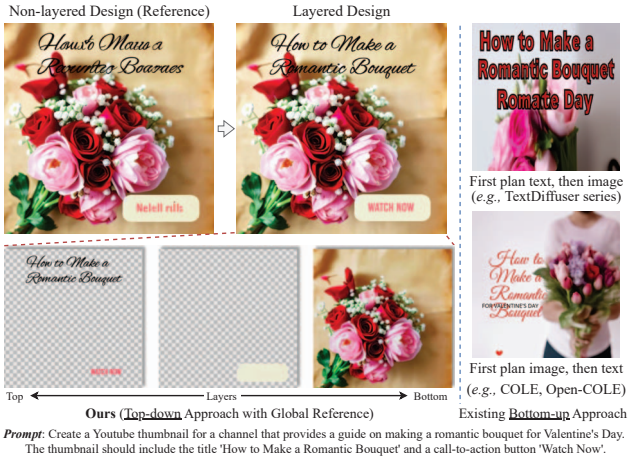


Figure 1. **Left:** We convert AI-generated non-layered design reference images to layered designs by extracting background, objects, and vectorized text layers in a top-down manner with optional further refinement. The layered representation facilitates easier editing. We include a gallery for more visualizations in Appendix A. **Right:** Existing bottom-up design approaches progressively generate elements on images in a predefined order, yet they lack a visual reference as the overall plan from the start to harmoniously orchestrate each element, often resulting in visual conflicts. For instance, text may occupy too much space, leaving inadequate room for the background, or causing overlaps between text regions.

erations directly applied on the layer representation.

Nevertheless, we still believe rasterized designs are of great value for creating layered designs, by realizing that human designers naturally use rasterized designs from textbooks or other sources as references to get inspiration in their layered design workflows. For instance, designers will use them in the beginning to explore suitable layouts, deciding where to place objects and what style of typography to use to achieve visual harmony. Given this practice, we aim to leverage rasterized designs as global references to create editable multi-layer designs, as illustrated in Figure 1 (left).

Based on this motivation, we introduce **Accordion**[1], a framework as demonstrated in Figure 2 built around a vision language model VLM consisting of three stages: (1) *reference creation*, (2) *design planning*, and (3) *layer generation*, while the VLM plays different roles in these stages. Notably, the framework can leverage a diverse and unlimited range of AI-generated references. It also offers the flexibility to start from the second stage when users explicitly provide designs as references. Besides, the VLM uses some vision experts in the design process. For example, SAM [27] and removal models [39] are used for element extraction and background filling. To the best of our knowledge, we take the **first attempt** to convert AI-generated designs to editable layered designs.

As shown in Figure 1 (right), in contrast to earlier bottom-up works, including text rendering methods such as TextDiffuser [5][2] that specifies the location of text before designing the background, and unlike COLE [22] that starts with the background to design text in the visual domain, our method starts with a global reference image representing the target design as a whole. This initial reference image globally orchestrates the layout and visual properties of various elements, ensuring overall visual harmony and preventing the conflicts that may arise in the step-by-step generation where earlier design decisions may not work well with later ones, as exemplified by generating text without leaving too much space for the foreground objects.

Accordion is trained using the in-house graphic design dataset Design39K. We innovatively use an inpainting model to mimic AI-generated design images and also obtain the refined text ground truth to construct the pairs. Experimental results and visualizations show the strong design capabilities of our method, especially its superior performance across various tasks on the DesignIntention benchmark [22], including text-to-template, adding text to background, and text de-rendering. Additionally, we explore the potential of Accordion to facilitate creative design variation, including using upstream generative models to modify references, applying inference time variations, and utilizing downstream generative models for further variations based on the extracted layers. Overall, our contributions can be summarized as follows:

- We rethink the layered design process by adopting a top-down approach, using the global reference to decompose each layer ensuring that all elements are harmonious. We take the first attempt to convert AI-generated designs into editable and practical layered designs.
- Accordion achieves superior performance on the DesignIntention benchmark in tasks such as text-to-template, adding text to backgrounds, and text de-rendering.
- Accordion enables a variety of variations and benefits from integration with existing generative models.

2. Related Work

Layered Design Generation There are a few investigations that focus on generating layered designs [20, 22, 42] considering the need for editability and customization. For instance, COLE [22] starts from a brief user-provided prompt, employing multiple large language models (LLMs) and diffusion models to generate each element within the design. Even though COLE uses language to comprehensively plan the design, it still visually constructs the design step-by-step, starting from the background, then generating objects, and finally the text. This sequential approach may lead to visual conflicts such as failing to allocate sufficient or suitable space for text or objects when generating background, often resulting from the lack of a global visual impression in mind. Open-COLE [20] adheres to the architecture of COLE but incorporates certain simplifications, such as omitting the object generation stage. Additionally, commercial tools such as CanvaGPT [3] utilize preset templates to adapt user prompts to new layered designs as noted in [22]. However, the diversity of designs is limited by the size of template libraries. We notice that a few methods [43, 50, 53] are designed for the layer generation of natural images. They are unsuitable for design images due to differences such as the absence of text layers in natural images. Another line of research involves predicting the attributes of partial elements given any existing elements, which are then combined together to create a layered design. Some works focus on predicting the layout of elements within a design, typically emphasizing the importance of proper alignment and the avoidance of overlaps to enhance visual aesthetics [1, 2, 4, 8, 13, 16, 17, 19, 24–26, 28–32, 40, 41, 46, 47, 49, 55, 58–60].

Overall, we first explore how to construct layered designs powered by the generative model and VLM, distinguishing it from existing methods.

[1]Our method is named "Accordion" because it unfolds rasterized designs into layered designs, similar to how an accordion expands. Besides, the proposed Accordion framework harmoniously integrates each element within a layered design, much like orchestrating every note harmoniously in a musical score at the concert.

[2]Text rendering methods usually require an additional text segmentation model and inpainting model for image layering.

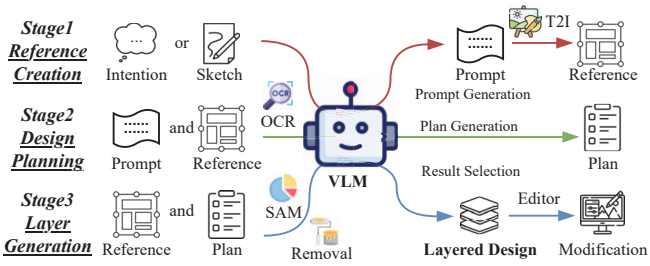


Figure 2. Overview of the proposed Accordion built around a VLM. It consists of three stages for constructing the design reference, plan, and layers respectively. Multiple vision experts (*e.g.,* OCR and SAM) are employed to create layered designs.

Non-layered Design Generation Existing GenAI methods are capable of creating an unlimited number of non-layered design images. Methods like Stable Diffusion [10, 38, 39] has demonstrated this capability of design image generation guided by user prompts. However, these methods often suffer from text rendering errors [9], which can significantly impact the usability and aesthetic of the generated images. To alleviate the text rendering issue, some works incorporate control over text areas during the image generation process [5, 6, 12, 21, 34–37, 44, 48, 52, 54, 56, 57]. While these methods improve the accuracy of text rendering, the final images are rasterized in pixel format, which prevents users from making edits easily.

Overall, despite the limitations of non-layered designs, we suppose these images are visually compelling and offer valuable references for the creation of layered designs.

3. Methodology

We sequentially detail the three stages of the proposed Accordion framework as illustrated in Figure 2, explaining the different roles the VLM plays in each stage. Note that we use AI-generated images as references for illustrative purposes. Accordion can also start from the later two stages if users explicitly provide other types of references such as an existing design or a background layer.

Problem formulation. Starting from an initial short intention I or sketch image S, we obtain an intermediate reference image R as global guidance, and expect output as a set of layers $\{O_1, O_2, \ldots, O_N, B, T\}$, where O_* represents object layers, where N is the total amount of foreground object layers. B is a background layer, and T is a vectorized text layer. Finally, these layers are stacked in the predicted order to form a layered design.

3.1. Stage1: Reference Creation

In this stage, our objective is to generate an image that serves as a global reference for the later two stages. Accordion is designed to accommodate a variety of inputs, including user-provided short intentions I or sketch drafts S, thus catering to diverse user preferences and needs. To facilitate this, we employ the VLM for **prompt generation**. As mentioned in COLE [22], users may provide only short intentions for simplicity, such as "*create a poster for Father's day*". We use in-context learning by supplying the VLM with multiple examples for prompt enhancement following Open-COLE [20]. This approach enables the VLM to generate detailed descriptive prompts P_{des} such as *"A father embraces his child in the center, surrounded by the text Father's Day and My All Time Hero."*. It is observed that detailed and lengthy prompts can facilitate the creation of images with rich detail [7] and significantly improve text quality [6] for better reference. If users wish to provide more layout constraints, they can use sketch drafts. In such cases, the VLM is instructed to give detailed descriptions of the depicted objects, accurately depicting object positions and filling text in suitable areas. We show the prompt templates and visualizations in Appendix B. Subsequently, these generated prompts are fed into a text-to-image (T2I) model to generate images as references R to be used in the next stage.

3.2. Stage2: Design Planning

In this stage, we aim to derive a design plan based on the rasterized reference image. This plan should detail the placement order of objects within the image to prepare for subsequent extraction, and provide the rendering attributes of text to facilitate the construction of vectorized text layers.

We employ VLM for **plan generation**. As illustrated in Figure 3, the VLM processes the reference image alongside a combined prompt P, which is concatenated by a predefined task description P_{task}, a description P_{des}, and OCR string P_{ocr}. The description is derived from the first stage detailed prompt, which is crucial for adapting textual content within the design plan, particularly for refining any nonsensical text produced by generative models. For OCR string, we use the text detection results of GenAI images, as the nonsensical content offers limited reference value. We show some cases of combined prompts in Appendix C.

As for the output, the VLM generates a sequence of dictionaries $\{D_*\}$, each representing the attributes of elements in the image, arranged in a bottom-to-top order to facilitate further object extraction. The output includes bounding boxes for both background and foreground objects and detailed text attributes such as bounding boxes, content, color, font, alignment, line count, and angle. Box coordinates are normalized to the range [0, 336]. For the color attributes (R, G, B, A), we map the [0, 255] range to [0, 25] to facilitate learning inspired by COLE [22]. Extending these attributes is straightforward and can be accomplished by incorporating additional fields into the training dataset. So far, this design plan forms the foundation for the next stage.

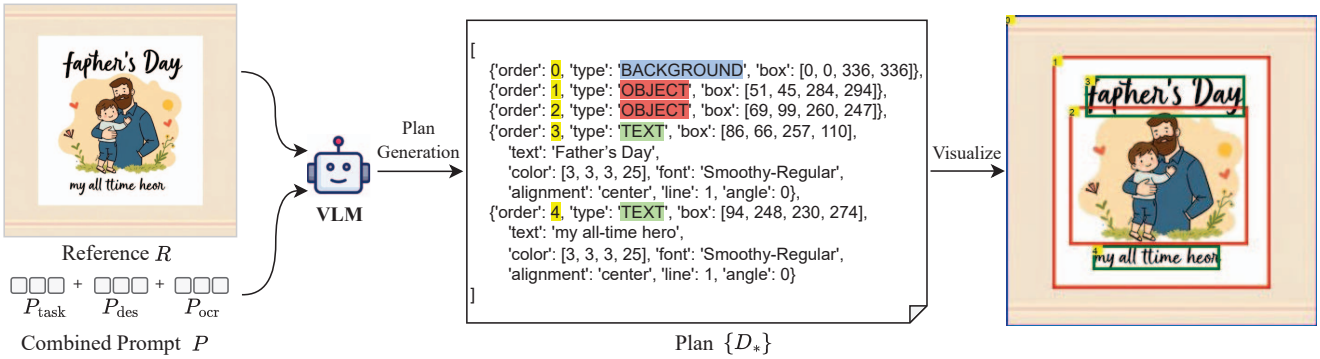
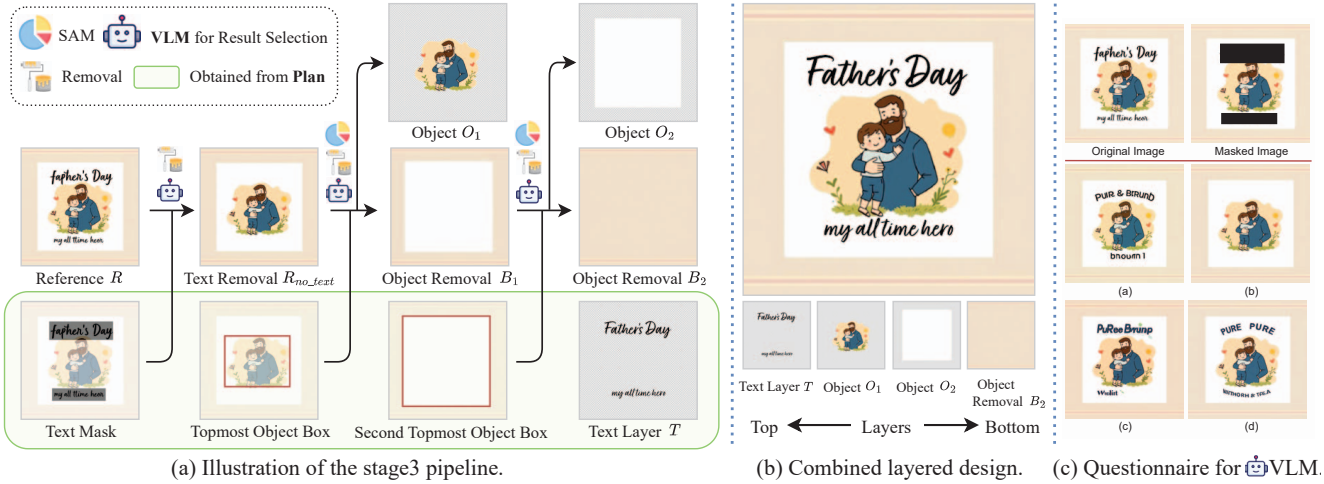


Figure 3. In Stage2 design planning, VLM processes the reference image alongside a combined prompt to generate a comprehensive design plan. This plan includes detailed information about each design element, including the background, objects, and text.



(a) Illustration of the stage3 pipeline. (b) Combined layered design. (c) Questionnaire for VLM.

Figure 4. Overview of the Stage3 layer generation. (a) Guided by the plan, the reference image is processed first with text removal, and then with progressive foreground object extraction using SAM and an object removal model, obtaining the background image in the end. (b) The text, objects, and background are stacked into a layered design. (c) The VLM conducts result selection using questionnaires throughout this process for better removal results.

3.3. Stage3: Layer Generation

In this stage, our goal is to construct the layered design. As illustrated in Figure 4, our approach is to extract and remove elements from the reference image, and then stack them back together to construct the final layered design. Although De-Render [42] also possesses the capability to extract layers from design images, it cannot extract object layers and lack the capability to refine nonsensical words.

Our initial step is to use a text removal model to erase text from the reference image R, recognizing that text regions are commonly placed on the top layer. Note that in the generated plan $\{D_*\}$, each text's bounding box is already included, which we can use as a condition to remove the text. Besides, the vectorized text layer T can be obtained from the plan, detailing the attributes of the text. Based on the text removal result R_{no_text}, our next focus moves to object removal, where we sequentially extract the topmost element according to the order outlined in the design plan. The object removal can be executed iteratively if multiple objects are detected in the design plan. In detail, for the nth object extraction, we use the SAM on the image B_{n-1} (when $n = 1$, the image R_{no_text} is used instead), which is conditioned on the bounding box from the design plan, to extract the foreground object O_n and its mask. The mask and the intermediate image are fed into an object removal model to remove the foreground object and obtain the intermediate background B_n. The last predicted background is used as the final background B.

It is noticed that the removal model sometimes generates diverse results, not all of which are satisfactory. To ensure consistent quality, we design a questionnaire that enables the VLM to conduct **result selection**, as illustrated in Figure 4 (c). The top row shows the original image alongside

Figure 5. For the same design, we utilize three types of references for the VLM, corresponding to three distinct training tasks: original designs, designs featuring nonsensical text, and designs with the text removed. We employ an inpainting model of varying strengths to generate images containing nonsensical text, where higher inpainting strength results in greater distortion of text details.

the masked image, where the text is masked with text boxes for the text removal task, and the object is masked using the SAM segmentation map for the object removal task. During the training phase, we present the VLM with the ground truth of the removal alongside three generated removal results, training it to select the highest quality option. In the inference stage, we generate four removal results for the VLM to choose the best one. We showcase the task prompt and some samples in Appendix D. So far, all the extracted objects $\{O_1, O_2, \ldots, O_n\}$ and the background B are combined with the text layer T rendered from the design plan to create a layered design.

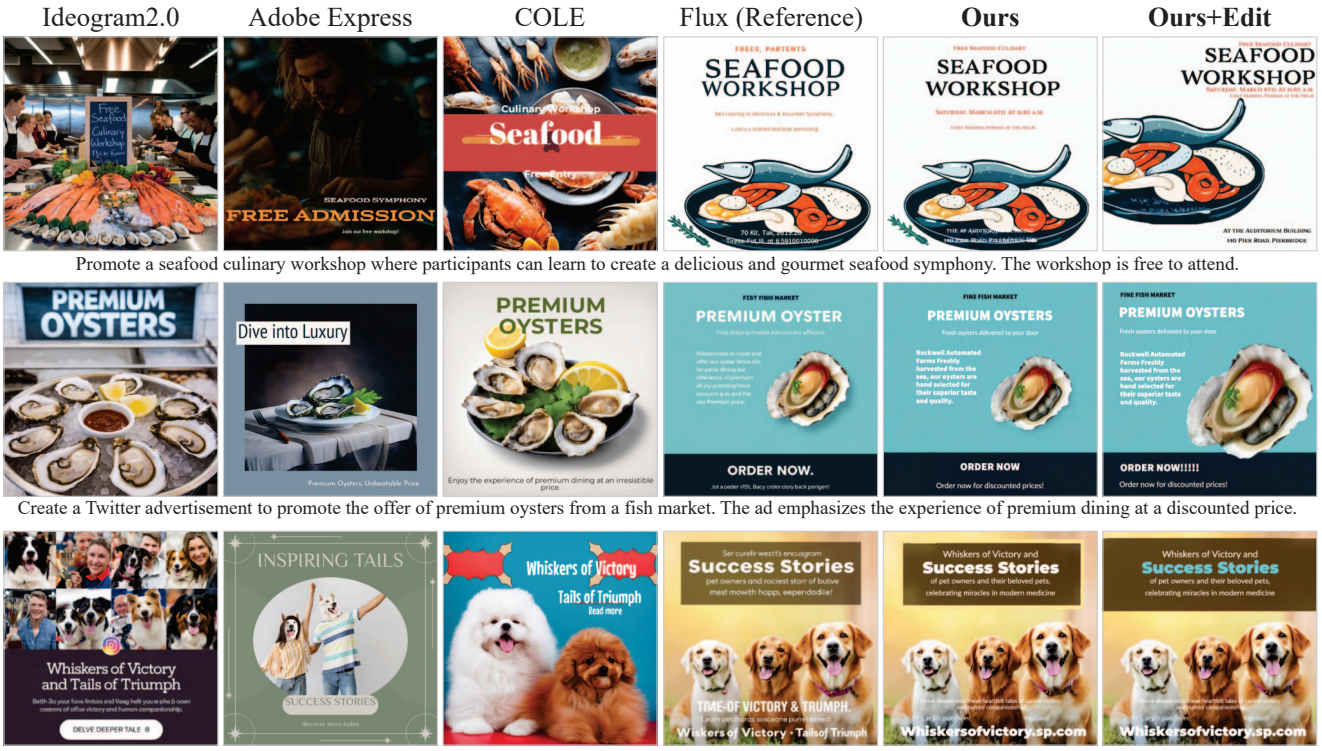
4. Experiments

4.1. Implementation Details

Dataset. We employ an in-house layered design dataset **Design39K**, with 39,233 samples for training and 492 samples for validation. This dataset comprises a diverse array of designs, including posters, book covers, advertisements, etc. Each sample is accompanied by a description. It is easy to extract layer information from these designs, including text, objects, and backgrounds, along with various attributes. We present more details about the dataset in Appendix E. As shown in Figure 5, to enrich the reference sources, we incorporate the following types of designs for training: (1) *Original designs*. We wish the model to conduct text derendering by directly parsing these original designs; (2) *Designs with nonsensical text*. We employ the Stable Diffusion 1.5 inpainting model [39] to inpaint text areas with inpainting strength randomly set between 0.5 and 0.7, leading to the generation of nonsensical text by the model. This range is selected because strength outside this range can lead to either insufficient or excessive inpainting changes, which hinder effective training. Please note that in some cases inpainting results may not strictly maintain the original text style, with potential variations in color and font. We consider this acceptable as it enables the model to use references while generating creative variations. The goal is to enable our model to effectively handle GenAI designs; (3) *Background without text*. By removing all text from a design and using only the background image as a reference, we challenge the model to add text in appropriate context and locations. We organize the elements of the design into a list of dictionaries and convert them into a string format for training the VLM. We employ the same training objectives for the three aforementioned reference types. Besides, to train the VLM for result selection, for both text and object removal tasks, we use the removal model to generate three different results. These results are then combined with the ground truth and randomly shuffled to construct the questionnaire dataset. In total, we have prepared 156,932 samples, consisting of 39,233 training samples for each of the three different types of references and the questionnaire.

Selection of vision expert models. For the Text-to-Image (T2I) model, we utilize Flux for its proficiency in generating high-quality references. Additionally, we use PaddleOCR as the OCR tool. SAM [27] and inpainting ControlNet [51] for Stable Diffusion 1.5 [39] with the prompt "nothing in the image" are used in the layer generation stage. Importantly, these models are modular and can be replaced when more advanced alternatives are available. We detail each expert model in Appendix F.

Training, inference, evaluation, and visualization. We use the vision language model LLaVA-1.5-7B [33] as the cornerstone of the proposed Accordion. The model is trained on the aforementioned 156,932 samples using LoRA [18] with learning rate 2e-4 for 6 epochs, conducted on 8 $\times$ 80G A100 GPUs for 36 hours. The reference image is scaled with the longer side set to 336 pixels following [33] for the VLM. The removal models operate at a resolution of 512$\times$512, which is the size of the final output. During inference, the average generation time per sample is 36.7 seconds (7, 19.5, 10.2 seconds on average for the three stages, respectively). For evaluation, we use the DesignIntention benchmark [22] which provides 500 detailed prompts across various design domains. We visualize the layered design through a Streamlit HTML frontend, with details provided in Appendix G.



Promote a seafood culinary workshop where participants can learn to create a delicious and gourmet seafood symphony. The workshop is free to attend.

Create a Twitter advertisement to promote the offer of premium oysters from a fish market. The ad emphasizes the experience of premium dining at a discounted price.

Create an Instagram advertisement showcasing success stories of pet owners and their purebred dogs. Encourage viewers to learn more about stories of victory and triumph.

Figure 6. Qualitative results of text-to-template show that our method uses references generated by Flux to create layered designs where elements appear more harmonious, and supports layered editing compared with Ideogram2.0. Intentions are shown below each image.

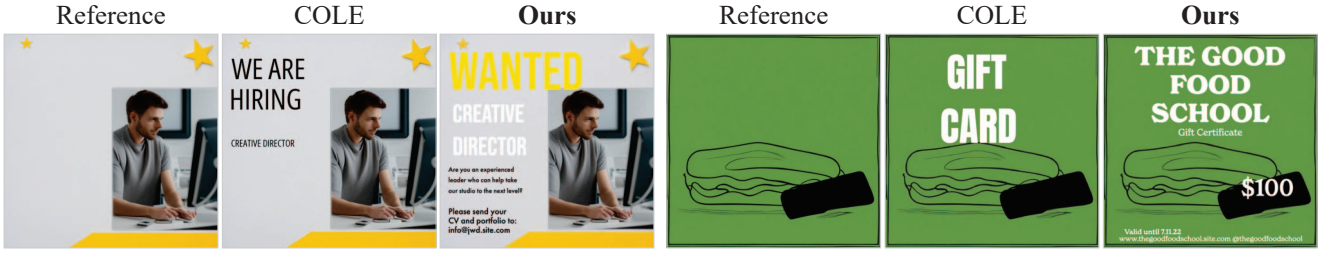


Figure 7. Qualitative results of adding text to the background show that our method not only places text harmoniously in terms of style and layout but also effectively utilizes space to create more complex and aesthetically pleasing designs than COLE.

4.2. Experimental Results

Text-to-template. As shown in Figure 6, we generate reference images using Flux [11] and compare them with commercial tools such as Ideogram2.0 and Adobe Express, as well as with images generated by COLE [22]. Some samples are shown in Figure 6. It is noteworthy that although the images produced by Ideogram2.0 appear fancy, they lack editability, and some text areas exhibit artifacts. Adobe Express produces vector designs with professional styles, but the generated text does not always match the input query closely. For example, in the second row, we intend to generate an advertisement for an "oyster discount", but the output contains a text saying "dive into luxury". COLE can generate layered designs, but the elements lack visual harmony. For example, in the first row, the placement of "Culinary Workshop" and "Free Entry" looks unnatural; and in the third row, the positioning of two red objects conflicts with the foreground text. Our method uses Flux to create references. We observe that due to the exceptional generative capabilities of Flux, the rasterized images it produces feature visually harmonious elements, despite the presence of nonsensical text. Then our method extracts objects from these references, learns the text style, and improves upon nonsensical text areas in the references, demonstrating superior performance. We also showcase editing results in the last column, including operations on text and objects. It is

Figure 8. Qualitative results of text de-rendering. Our method groups text at the paragraph level to predict their attributes, thereby enhancing visual harmony. Through the translation to Chinese task, we better maintain semantic coherence, style harmony, and spatial alignment compared with De-Render, which predicts the attributes of each individual word, resulting in an overall appearance that lacks harmony. We also translate the design from Chinese back to English to verify its consistency with the original design. Our approach significantly surpasses the original in terms of semantic and layout consistency.

worthy of note that our method is adaptable to reference images generated by various models, such as substituting Flux with SD3 [10] or future advanced models. In contrast, COLE relies on fine-tuning a base generative model to achieve layered designs, necessitating retraining to adapt to newer generative models, which diminishes its compatibility. Furthermore, the proposed Accordion supports the generation of designs with varying aspect ratios, whereas COLE is limited to producing only square designs. These results are detailed in the Appendix H.

We validate the quantitative results using the protocol provided by COLE, which evaluates performance across five metrics. These metrics are scored via preset prompt queries to GPT-4V, with scores ranging from 0 to 10 and higher scores indicate better performance. Our comparisons are primarily with academic methods following [20] because commercial tools typically lack open-source code or accessible APIs for large-scale testing. As shown in Table 1, our method achieves the highest average score (6.5) compared to existing layered approaches such as COLE and Open-COLE. It is worth noting that our method outperforms COLE, despite the fact that COLE uses a 100K in-house dataset for training, which is far more extensive than our dataset of 39K. In addition, we specifically evaluate the content relevance of images generated by Flux and our method. The experimental results show that our method outperforms Flux by 0.4 in this metric (7.0 for Flux *v.s.* 7.4 for ours). This improvement demonstrates that our method refines the image by replacing nonsensical text with relevant information, thereby enhancing the content relevance of text layers to backgrounds. We also employ an aesthetic score for evaluation. Specifically, we utilize LAION's aesthetic tool to assess the aesthetic quality of the designs. Our method achieved a high score of 4.98, outperforming COLE's method, which scored 4.72.

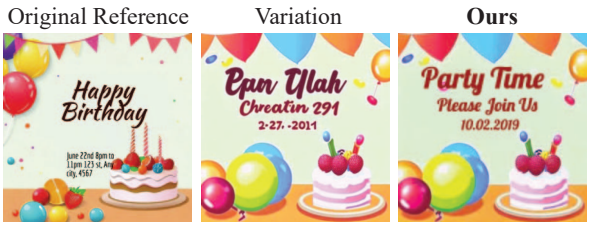
Adding text to background. Since COLE provides the SVG-format results for the DesignIntention benchmark [22], we can easily remove the text to obtain the background. As illustrated in Figure 7, we compare our method

Table 1. Quantitative results for the comparison of layered design methods using the DesignIntention benchmark. The five metrics are: (i) design and layout, (ii) content relevance, (iii) typography and color, (iv) graphics and images, and (v) innovation. *Open-COLE cannot generate the foreground object layers. The proposed Accordion achieves the best average performance.

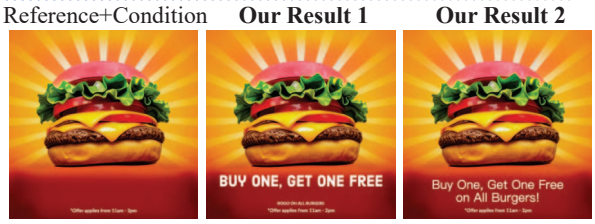
Methods	(i)	(ii)	(iii)	(iv)	(v)	Avg.
COLE [22]	6.0	6.9	5.7	6.2	5.1	6.0
Open-COLE* [20]	6.3	7.0	5.6	7.1	**5.3**	6.3
Accordion (Ours)	**6.7**	**7.4**	**6.1**	**7.3**	5.1	**6.5**

with the results from COLE. The results demonstrate that our method achieves good harmony in both layout and text style. In addition, we observe that COLE tends to produce layouts that are relatively simple, often avoiding longer sentences. Through analysis, it is evident that our model is able to generate text with more informative content. We conclude that the reason for this issue is that COLE predetermines the text content without considering the available space in the background image, potentially resulting in a disruption of visual harmony. Our samples show a greater overall text length, averaging 61.7 characters compared to 42.3 characters for COLE (approximately 1.5 times longer). In other words, our approach can fully utilize empty space in the background to increase information density and enhance overall aesthetics. This makes our model especially well-suited for scenarios like detailed reports and comprehensive advertisements. We quantitatively compare our results with COLE using GPT-4V, where images generated by both methods are concatenated horizontally and shown to GPT-4V to assess which one has higher quality. The task prompt is provided in Appendix I. The results show a preference for our model on 52% of all the samples compared to 48% for COLE.

De-rendering. In Figure 8, we validated the text de-rendering capability of our model in comparison with De-Render [42]. Note that our method treats paragraph-level text as a single entity. One advantage of paragraph-level



(a) We use an image variation model to create similar-style design variations, which are then refined by our model.



(b) Our model can generate variations conditioned on the text layer conditioned on existing text on the background

(c) The extracted layers can be used for downstream layout generation models for creating variations.

Figure 9. Our methods can support the design variations.

representation is the uniformity of style, a feature that De-Render struggles to achieve. In contrast, De-Render predicts a style for each word individually, which can lead to a visually disorganized appearance. Moreover, we showcase our advantage with a translation application. By grouping words at the paragraph level, our approach effectively considers the coherence of sentence semantics during translation, ensuring consistent style and alignment of adjacent words. Since De-Render operates at the word level, it lacks overall contextual information in translation application, leading to disorganized layouts with overlapping text.

Another critical evaluation metric is the ability to de-render the correct number of layers from a rasterized design. We test on the detection of layer quantities using the Design39K evaluation set. The mean absolute error (L1 loss) between detected values and ground truth, is 0.494 for text layers and 0.274 for object layers. This indicates that the detection of layer numbers is relatively accurate.

Design variation creation. As shown in Figure 9, we demonstrate three methods based on our approach for creating design variations under three categories: (a) *Variation by upstream models* based on an image variation model [45]. Typically, image editing models such as variations models or inpainting models tend to degrade the text regions within images. Our method is capable of refining these areas. (b) *Variation at inference time*. The VLM can generate different prediction branches based on the existing text on the background, corresponding to various designs. (C) *variation by downstream models* based on VLC [41]. Our model is capable of decomposing existing designs for downstream layout generation models. Overall, these methods demonstrate the versatility and critical role of our approach in the design workflow.

4.3. User Studies by Designers

To evaluate the *editability* of the layered design and to differentiate it from COLE and Open-COLE, which rely solely on image-level evaluation, we enlisted 29 designers to assess 30 cases. In the text-to-template editability evaluation, 73.5% of the cases were deemed superior to those of COLE. Furthermore, in the sketch-to-description task, 87.2% of the cases were considered appropriate. These higher scores demonstrate that users can effectively operate the system with straightforward inputs, thereby significantly reducing complexity. Detailed results of the user studies are presented in Appendix J.

4.4. Ablation Studies

Multi-task training for VLM. Here we investigate whether joint training gains benefits or leads to degradation. The results for each task are presented in Appendix K. We observe that joint training and separate training yield comparable average scores, with joint training slightly outperforming by a margin of 0.82%. We opt for a more compact model architecture and ultimately choose joint training.

Questionnaire selection effect. We evaluate the efficacy of using questionnaires for the result selection process. The results indicate that for text removal tasks, the PSNR increases from 18.01 to 21.18, and for object removal tasks, it improves from 22.92 to 24.87. It demonstrates the effectiveness of this approach.

5. Conclusions and Discussions

We introduce Accordion, which takes the first attempt to convert AI-generated designs to editable designs. It has three stages and uses reference images to create layered designs. We have employed plenty of evaluation metrics to assess the performance of the proposed method, including the GPT evalaution score, aesthetic score, as well as the evaluation by human designers to rate the editability. It shows superior performance on the DesignIntention benchmark and we fully explore the potential of Accordion to create the design variations. For the *limitations*, the SAM model attained an IOU score of 68.4%, occasionally facing challenges with hollow parts in objects and incomplete extractions. These issues often stem from cumulative errors in upstream VLM detection, including misaligned bounding boxes around the objects. For *future work*, we aim to employ VLM as an agent to integrate additional expert models.

6. Acknowledgement

This research was supported by the Research Grant Council of the Hong Kong Special Administrative Region under grant number 16212623.

References

[1] Sanket Biswas, Pau Riba, Josep Lladós, and Umapada Pal. Docsynth: a layout guided approach for controllable document image synthesis. In *ICDAR*, 2021. 2

[2] Sanket Biswas, Rajiv Jain, Vlad I Morariu, Jiuxiang Gu, Puneet Mathur, Curtis Wigington, Tong Sun, and Josep Lladós. Docsynthv2: A practical autoregressive modeling for document generation. *arXiv preprint arXiv:2406.08354*, 2024. 2

[3] canva. Link: https://www.canva.com/magic-write/, 2024. 2

[4] Shang Chai, Liansheng Zhuang, Fengying Yan, and Zihan Zhou. Two-stage content-aware layout generation for poster designs. In *ACMMM*, 2023. 2

[5] Jingye Chen, Yupan Huang, Tengchao Lv, Lei Cui, Qifeng Chen, and Furu Wei. Textdiffuser: Diffusion models as text painters. In *NeurIPS*, 2023. 1, 2, 3

[6] Jingye Chen, Yupan Huang, Tengchao Lv, Lei Cui, Qifeng Chen, and Furu Wei. Textdiffuser-2: Unleashing the power of language models for text rendering. In *ECCV*, 2024. 3

[7] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, et al. Pixart-alpha: Fast training of diffusion transformer for photorealistic text-to-image synthesis. In *ICLR*, 2024. 3

[8] Yutao Cheng, Zhao Zhang, Maoke Yang, Hui Nie, Chunyuan Li, Xinglong Wu, and Jie Shao. Graphic design with large multimodal model. *arXiv preprint arXiv:2404.14368*, 2024. 2

[9] Giannis Daras and Alexandros G Dimakis. Discovering the hidden vocabulary of dalle-2. *arXiv preprint arXiv:2206.00169*, 2022. 3

[10] 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 *ICML*, 2024. 3, 7

[11] flux. Link: https://blackforestlabs.ai/#get-flux, 2024. 6

[12] Yifan Gao, Jinpeng Lin, Min Zhou, Chuanbin Liu, Hongtao Xie, Tiezheng Ge, and Yuning Jiang. Textpainter: Multimodal text image generation with visual-harmony and text-comprehension for poster design. In *ACMMM*, 2023. 3

[13] Kamal Gupta, Justin Lazarow, Alessandro Achille, Larry S Davis, Vijay Mahadevan, and Abhinav Shrivastava. Layouttransformer: Layout generation and completion with self-attention. In *ICCV*, 2021. 2

[14] Yingqing He, Zhaoyang Liu, Jingye Chen, Zeyue Tian, Hongyu Liu, Xiaowei Chi, Runtao Liu, Ruibin Yuan, Yazhou Xing, Wenhai Wang, et al. Llms meet multimodal generation and editing: A survey. *arXiv preprint arXiv:2405.19334*, 2024. 1

[15] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. In *ICLR*, 2023. 1

[16] Daichi Horita, Naoto Inoue, Kotaro Kikuchi, Kota Yamaguchi, and Kiyoharu Aizawa. Retrieval-augmented layout transformer for content-aware layout generation. In *CVPR*, 2024. 2

[17] Hsiao Yuan Hsu, Xiangteng He, Yuxin Peng, Hao Kong, and Qing Zhang. Posterlayout: A new benchmark and approach for content-aware visual-textual presentation layout. In *CVPR*, 2023. 2

[18] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *ICLR*, 2022. 5

[19] Naoto Inoue, Kotaro Kikuchi, Edgar Simo-Serra, Mayu Otani, and Kota Yamaguchi. Towards flexible multi-modal document models. In *CVPR*, 2023. 2

[20] Naoto Inoue, Kento Masui, Wataru Shimoda, and Kota Yamaguchi. Opencole: Towards reproducible automatic graphic design generation. *arXiv preprint arXiv:2406.08232*, 2024. 2, 3, 7

[21] Jiabao Ji, Guanhua Zhang, Zhaowen Wang, Bairu Hou, Zhifei Zhang, Brian Price, and Shiyu Chang. Improving diffusion models for scene text editing with dual encoders. *Transactions on Machine Learning Research (TMLR)*, 2023. 3

[22] Peidong Jia, Chenxuan Li, Zeyu Liu, Yichao Shen, Xingru Chen, Yuhui Yuan, Yinglin Zheng, Dong Chen, Ji Li, Xiaodong Xie, et al. Cole: A hierarchical generation framework for graphic design. *arXiv preprint arXiv:2311.16974*, 2023. 2, 3, 5, 6, 7

[23] Yueru Jia, Yuhui Yuan, Aosong Cheng, Chuke Wang, Ji Li, Huizhu Jia, and Shanghang Zhang. Designedit: Multi-layered latent decomposition and fusion for unified & accurate image editing. *arXiv preprint arXiv:2403.14487*, 2024. 1

[24] Zhaoyun Jiang, Shizhao Sun, Jihua Zhu, Jian-Guang Lou, and Dongmei Zhang. Coarse-to-fine generative modeling for graphic layouts. In *AAAI*, 2022. 2

[25] Akash Abdu Jyothi, Thibaut Durand, Jiawei He, Leonid Sigal, and Greg Mori. Layoutvae: Stochastic scene layout generation from a label set. In *ICCV*, 2019.

[26] Kotaro Kikuchi, Naoto Inoue, Mayu Otani, Edgar Simo-Serra, and Kota Yamaguchi. Multimodal markup document models for graphic design completion. *arXiv preprint arXiv:2409.19051*, 2024. 2

[27] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *ICCV*, 2023. 2, 5

[28] Chenhui Li, Peiying Zhang, and Changbo Wang. Harmonious textual layout generation over natural images via deep aesthetics learning. *IEEE Transactions on Multimedia (TMM)*, 2021. 2

[29] Jianan Li, Jimei Yang, Aaron Hertzmann, Jianming Zhang, and Tingfa Xu. Layoutgan: Generating graphic layouts with wireframe discriminators. In *ICLR*, 2019.

[30] 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.

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

[32] 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 *ACMMM*, 2023. 2

[33] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023. 5

[34] 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 *ECCV*, 2024. 3

[35] 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.

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

[37] 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. 3

[38] 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 *ICLR*, 2023. 3

[39] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 2, 3, 5

[40] Jaejung Seol, Seojun Kim, and Jaejun Yoo. Posterllama: Bridging design ability of langauge model to contents-aware layout generation. *arXiv preprint arXiv:2404.00995*, 2024. 2

[41] Mohammad Amin Shabani, Zhaowen Wang, Difan Liu, Nanxuan Zhao, Jimei Yang, and Yasutaka Furukawa. Visual layout composer: Image-vector dual diffusion model for design layout generation. In *CVPR*, 2024. 2, 8

[42] Wataru Shimoda, Daichi Haraguchi, Seiichi Uchida, and Kota Yamaguchi. De-rendering stylized texts. In *ICCV*, 2021. 2, 4, 7

[43] Petru-Daniel Tudosiu, Yongxin Yang, Shifeng Zhang, Fei Chen, Steven McDonagh, Gerasimos Lampouras, Ignacio Iacobacci, and Sarah Parisot. Mulan: A multi layer annotated dataset for controllable text-to-image generation. In *CVPR*, 2024. 2

[44] Yuxiang Tuo, Wangmeng Xiang, Jun-Yan He, Yifeng Geng, and Xuansong Xie. Anytext: Multilingual visual text generation and editing. In *ICLR*, 2024. 3

[45] Xingqian Xu, Zhangyang Wang, Gong Zhang, Kai Wang, and Humphrey Shi. Versatile diffusion: Text, images and variations all in one diffusion model. In *ICCV*, 2023. 8

[46] Kota Yamaguchi. Canvasvae: Learning to generate vector graphic documents. In *ICCV*, 2021. 2

[47] Ning Yu, Chia-Chih Chen, Zeyuan Chen, Rui Meng, Gang Wu, Paul Josel, Juan Carlos Niebles, Caiming Xiong, and Ran Xu. Layoutdetr: detection transformer is a good multimodal layout designer. *arXiv preprint arXiv:2212.09877*, 2022. 2

[48] Weichao Zeng, Yan Shu, Zhenhang Li, Dongbao Yang, and Yu Zhou. Textctrl: Diffusion-based scene text editing with prior guidance control. In *NeurIPS*, 2024. 3

[49] Junyi Zhang, Jiaqi Guo, Shizhao Sun, Jian-Guang Lou, and Dongmei Zhang. Layoutdiffusion: Improving graphic layout generation by discrete diffusion probabilistic models. In *ICCV*, 2023. 2

[50] Lvmin Zhang and Maneesh Agrawala. Transparent image layer diffusion using latent transparency. *ACM TRANSACTIONS ON GRAPHICS (TOG)*, 2024. 2

[51] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *ICCV*, 2023. 5

[52] Lingjun Zhang, Xinyuan Chen, Yaohui Wang, Yue Lu, and Yu Qiao. Brush your text: Synthesize any scene text on images via diffusion model. In *AAAI*, 2024. 3

[53] Xinyang Zhang, Wentian Zhao, Xin Lu, and Jeff Chien. Text2layer: Layered image generation using latent diffusion model. *arXiv preprint arXiv:2307.09781*, 2023. 2

[54] Qilong Zhangli, Jindong Jiang, Di Liu, Licheng Yu, Xiaoliang Dai, Ankit Ramchandani, Guan Pang, Dimitris N Metaxas, and Praveen Krishnan. Layout-agnostic scene text image synthesis with diffusion models. In *CVPR*, 2024. 3

[55] Nanxuan Zhao, Ying Cao, and Rynson WH Lau. Modeling fonts in context: Font prediction on web designs. In *Computer Graphics Forum*, 2018. 2

[56] Yiming Zhao and Zhouhui Lian. Udifftext: A unified framework for high-quality text synthesis in arbitrary images via character-aware diffusion models. In *ECCV*, 2024. 3

[57] Zhen Zhao, Jingqun Tang, Binghong Wu, Chunhui Lin, Shu Wei, Hao Liu, Xin Tan, Zhizhong Zhang, Can Huang, and Yuan Xie. Harmonizing visual text comprehension and generation. In *NeurIPS*, 2024. 3

[58] Xinru Zheng, Xiaotian Qiao, Ying Cao, and Rynson WH Lau. Content-aware generative modeling of graphic design layouts. *ACM Transactions on Graphics (TOG)*, 2019. 2

[59] Min Zhou, Chenchen Xu, Ye Ma, Tiezheng Ge, Yuning Jiang, and Weiwei Xu. Composition-aware graphic layout gan for visual-textual presentation designs. *arXiv preprint arXiv:2205.00303*, 2022.

[60] Wanrong Zhu, Jennifer Healey, Ruiyi Zhang, William Yang Wang, and Tong Sun. Automatic layout planning for visually-rich documents with instruction-following models. *arXiv preprint arXiv:2404.15271*, 2024. 2