Relation-Aware Diffusion Model for Controllable Poster Layout Generation

Fengheng Li[*][†]
TKLNDST, CS
Nankai University
Tianjin, China
lifengheng@foxmail.com

An Liu[*]
Retail Platform Operation
and Marketing Center, JD
Beijing, China
liuan39@jd.com

Wei Feng
Retail Platform Operation
and Marketing Center, JD
Beijing, China
fengwei25@jd.com

Honghe Zhu
Retail Platform Operation
and Marketing Center, JD
Beijing, China
zhuhonghe1@jd.com

Yaoyu Li
Retail Platform Operation
and Marketing Center, JD
Beijing, China
liyaoyu1@jd.com

Zheng Zhang
Retail Platform Operation
and Marketing Center, JD
Beijing, China
zhangzheng11@jd.com

Jingjing Lv
Retail Platform Operation
and Marketing Center, JD
Beijing, China
lvjingjing1@jd.com

Xin Zhu
Retail Platform Operation
and Marketing Center, JD
Beijing, China
zhuxin3@jd.com

Junjie Shen
Retail Platform Operation
and Marketing Center, JD
Beijing, China
shenjunjie@jd.com

Zhangang Lin
Retail Platform Operation
and Marketing Center, JD
Beijing, China
linzhangang@jd.com

Jingping Shao
Retail Platform Operation
and Marketing Center, JD
Beijing, China
shaojingping@jd.com

ABSTRACT

Poster layout is a crucial aspect of poster design. Prior methods primarily focus on the correlation between visual content and graphic elements. However, a pleasant layout should also consider the relationship between visual and textual contents and the relationship between elements. In this study, we introduce a relation-aware diffusion model for poster layout generation that incorporates these two relationships in the generation process. Firstly, we devise a visual-textual relation-aware module that aligns the visual and textual representations across modalities, thereby enhancing the layout's efficacy in conveying textual information. Subsequently, we propose a geometry relation-aware module that learns the geometry relationship between elements by comprehensively considering contextual information. Additionally, the proposed method can generate diverse layouts based on user constraints. To advance research in this field, we have constructed a poster layout dataset named CGL-Dataset V2. Our proposed method outperforms state-of-the-art methods on CGL-Dataset V2. The data and code will be available at https://github.com/liuan0803/RADM.

CCS CONCEPTS

• **Computing methodologies** → **Neural networks**.

---

[*]Both authors contributed equally to this research.
[†]Work done during an internship at JD.com.

KEYWORDS

Poster layout generation, Diffusion model, Controllable generation, Relation-aware

1 INTRODUCTION

Poster layout generation aims to predict the position and category of graphic elements on the image, which is important for visual aesthetics and information transmission of posters. Due to the need to consider both graphic relationships and image compositions when creating high-quality poster layouts, this challenging task is usually completed by professional designers. However, manual design is often time-consuming and financially burdensome.

To generate high-quality poster layouts at low cost, automatic layout generation has become increasingly popular in academia and industry. With the advent of deep learning, some content-agnostic methods [9, 10, 12, 13, 15, 30] are proposed to learn the internal relationship of graphic elements. However, these methods prioritize the graphic relationships between elements and overlook the impact of visual content on poster layout. Therefore, applying these methods directly to poster layout generation can negatively impact subject presentations, text readability and the visual balance of the poster as a whole. To address these issues, several content-aware methods [4, 16, 34] generate layouts based on the visual contents of input background images. ContentGAN [16] leverages visual and textual semantic information to implicitly model layout structures

Figure 1: The visual examples of poster layout produced by CGL-GAN[34] and ours.

and design principles, resulting in plausible layouts. However, ContentGAN lacks spatial information. To overcome this limitation, CGL-GAN [34] combines a multi-scale CNN and a transformer to extract not only global semantics but also spatial information, enabling better learning of the relationship between images and graphic elements.

Despite their promising results, two relationships still require consideration in poster layout generation. On one hand, text plays an important role in the information transmission of posters, so the poster layout generation should also consider the relationship between text and vision. As shown in the first row in Fig. 1, ignoring text during layout generation will result in the generated layout not being suitable for filling the given text content. On the other hand, a good layout not only needs to consider the position of individual elements, but also the coordination relationship between elements. As shown in the second row in Fig. 1, considering the geometric relationships between elements can work better on graphic metrics.

In this paper, we propose a relation-aware diffusion model for poster layout generation as depicted in Fig. 3, considering both visual-textual and geometry relationships. As diffusion models have achieved great success in many generation tasks [1, 2, 26, 32], we follow the noise-to-layout paradigm to generate poster layout by gradually adjusting noisy layout via the learned denoising model. In each sampling step, given a set of boxes sampled in Gaussian distribution or the estimated boxes from the last sampling step as input, we extract RoI features from the feature map generated by the image encoder. Then a Visual-Textual Relation-Aware Module (VTRAM) is proposed to model the relationship between visual and textual features, which makes the layout result determined by both the image and text content. Meanwhile, we design a Geometry Relation-Aware Module (GRAM) to enhance the features of each RoI based on its relative position to other RoIs. This enables the model to better understand the contextual information of graphic elements. Finally, the position and category of elements are determined by the outputs of VTRAM and GRAM, as well as the RoI features. The predicted results are sent to the next step to progressively refine themselves. Benefiting from the newly proposed VTRAM

and GRAM, users can regulate the layout generation process by predefining layouts or adjusting text content.

To summarize, the contributions of our work are listed below:

- We propose a novel visual-textual relation-aware module to study the relationship between visual and textual information, which makes the generated layout results easier for posters to convey text information.
- A geometry relation-aware module is used to explicitly learn the geometric relationships between elements, so that each element can consider the context more comprehensively.
- To promote research in this field, we extend the dataset proposed in CGL-GAN [34] to CGL-Dataset V2 by adding text content annotations. Extensive experiments show that our method outperforms state-of-the-art methods, and can generate layout based on user constraints.

2 RELATED WORK

2.1 Layout Generation

In recent years, there has been a surge of interest in the field of layout generation. Researchers have been exploring new techniques and algorithms to automate the process of designing layouts for various applications, such as web design [14, 22], graphic design [3, 33, 34], and even interior design [31]. Various techniques have been proposed to generate layouts automatically that are visually appealing and semantically meaningful. Prior approaches can be roughly divided into two subcategories: rule-based and template-based methods. Rule-based methods [3, 21, 22] define a set of rules that govern the placement of various elements in a layout. These rules are based on design principles and heuristics that have been established by experts in the field. Template-based methods [11, 24] involve using pre-defined templates to generate layouts that conform to specific design patterns. However, the methods mentioned above require professional knowledge and the generated layouts usually lack diversity. According to whether the visual content is considered, we divide the deep generative models into two categories: content-agnostic and content-aware methods. Content-agnostic methods usually yield layouts with visual balance and symmetry as there are fewer constraints, making them suitable for documents, user interfaces, and publication generation. Layout-VAE [12], which utilizes Variational Autoencoders, is a method that learns to produce layouts based on the categories of elements. To further improve the quality of the generated layouts, transformers [13, 30] are used in the generation task. Due to the attention mechanism, transformer-based methods are capable of implicitly learning the relationships between elements.

Nonetheless, content-agnostic methods tend to have inadequate performance when it comes to layout generation tasks that require comprehension of given content. To solve the problem, content-aware methods are proposed for specific tasks. ContentGAN [33] is the first model to incorporate both visual and textual semantics in the generation of magazine layouts. It used Generative Adversarial Networks (GANs) to learn complicated layout structures and generate layouts from noise, which enables the diversity of layouts. However, the lack of spatial information and detailed features of the image leads to unsatisfactory layout results under complex background conditions. More recently, transformer-based models such

Figure 2: (a) Poster layout annotation. Different colors represent different element types, the text annotation results are in the gray box, and the English translation is in brackets; (b) Clean image; (c) Input for inference stage.

as CGL-GAN [34] and LCVT [4] have been introduced for stronger layout capabilities. Although these methods introduce spatial visual information and domain alignment information respectively, they do not consider the impact of text content on layout and how to more accurately model the positional relationship between layout elements. Different from the above methods, we introduce visual and textual prior knowledge to generate layouts and consider geometric relation priors to strengthen the feature expression between layout elements.

2.2 Diffusion Models

In recent years, diffusion models [8, 27] have gradually become the focus of generative tasks because of their impressive high-quality generative capabilities. The diffusion and denoising processes are key components of this approach. Diffusion refers to the gradual transformation of an initial image into a final noisy image through a series of small, random perturbations. Denoising, on the other hand, is the process of learning to remove noise from the image to actual distribution. Besides image generation, Diffusion models are gaining momentum in various fields and showing promising performance. DiffusionDet [5] is the first to apply diffusion model for the task of object detection. InST [32] implemented Inversion-Based Style Transfer with Diffusion Models. Video LDM [2] achieved high-resolution video generation by training a diffusion model in a compressed low-dimensional latent space. Naturally, the diffusion model is also introduced into the field of layout generation. LayoutDM [10] uses a discrete diffusion model to predict the attributes of elements like category and position. LDGM [9] unifies unconditional and conditional generation in a single diffusion model. But these methods are oblivious to input contents and perform poorly in poster layout generation. By introducing a multimodal diffusion model, our method can align the image and texts and produce more visually convincing posters.

3 CGL-DATASET V2

CGL-Dataset V2 is a dataset for the task of automatic graphic layout design of advertising posters, containing 60,548 training samples

and 1035 testing samples. It is an extension of CGL-Dataset [34]. The original CGL-Dataset contains 4 types of elements: logos, texts, underlays and embellishments as shown in Fig. 2 (a). Each element consists of category and coordinates information. However, it does not include text content annotations, which have a crucial impact on the layout of posters. As shown in Fig. 2 (a), to study the influence of content, we supplementally annotate the textual content. In the training set, in order to obtain a clean background image for model training, we use an inpainting model [28] to erase layout elements, and the result is shown in Fig. 2 (b). The text information is not provided in the test set of the original CGL-Dataset, so we additionally collect 1035 poster images with usable textual descriptions to replace the original test set. As shown in Fig. 2 (c), the collected poster images are processed the same as the training set to get a clean background image. Meanwhile, we collected all the promotional slogans of the current product for analysis of different textual content for poster layout impact. Since the collected text content is more focused on the e-commerce field, we use a pretrained model based on massive e-commerce text corpus training to extract textual features. The extraction method is detailed in section 4.2. For convenience, we will publish the language model for extracting textual features.

4 METHOD

The overview of our method is shown in Fig. 3. The proposed method is composed of four parts: feature extractor, Visual-Textual Relation-Aware Module (VTRAM), Geometry Relation-Aware Module (GRAM) and layout decoder. The feature extractor extracts features from text and images respectively. Then VTRAM models the visual and textual relationship for superior layouts. Meanwhile, GRAM is used to strengthen the ability to express the positional relationship between each RoI feature. Finally, based on the outputs of VTRAM and GRAM, as well as the RoI features, the layout decoder predicts the coordinates and category of elements. Next, we will introduce the process of applying the diffusion mechanism to poster layout generation and the details of the four parts.

Figure 3: The overview of our method, which contains four parts: feature extractor, VTRAM, GRAM and layout decoder.

Figure 4: Inspired by diffusion denoising process, from left to right, we formulate the poster layout generation as a process to gradually refine the position and size of boxes from step T to step i.

4.1 Poster Layout Generation with Diffusion Model

Diffusion models are a class of probabilistic generative models that convert noise to a representative data sample by using Markovian chain. As shown in Fig. 4, we formulate the poster layout generation problem as a noise-to-layout generative process by gradually adjusting the noise layout with a learned denoising model. The poster layout generated by the diffusion model also includes two processes: the diffusion process and the denoising process. Given a poster layout, we gradually add Gaussian noise to corrupt the deterministic layout result, we call this operation the diffusion process. Instead, given an initial random layout, we obtain the final poster layout by stepwise denoising, which is called the denoising process. Next, we will introduce the diffusion process and the denoising process respectively.

4.1.1 Diffusion Process. x_0 is a set of layout elements, each element consists of coordinates (x, y, w, h), where x, y, w, h represent the horizontal center, vertical center, width and height of the rectangular box, respectively. We get sample data x_0 from a true data distribution $q(x)$ and gradually add Gaussian noise to sample data in each step i. We get a sequence of intermediate samples $x_1, \cdots, x_i, \cdots, x_T$. The noise is controlled by the variance schedule

$\beta(\beta_i \in (0, 1))$.

$$q(x_i|x_{i-1}) = \mathcal{N}(x_i; \sqrt{1 - \beta_i}x_{i-1}, \beta_i\mathbf{I}),$$
$$q(x_{1:T}|x_0) = \prod_{i=1}^{T} q(x_i|x_{i-1}). \tag{1}$$

With the nice property found by [8], we can directly sample x_i at any arbitrary time step i as:

$$q(x_i|x_0) = \mathcal{N}(x_i; \sqrt{\hat{\alpha}_i}x_0, (1 - \hat{\alpha}_i)\mathbf{I}),$$
$$\hat{\alpha}_i = \prod_{j=1}^{i}(1 - \beta_j). \tag{2}$$

4.1.2 Denoise Process. These conditional probabilities $q(x_{i-1}|x_i)$, however, are intractable. Instead, we train a model $f_\theta(t, x_t, I_{img}, I_{text})$ to approximate the reverse process, where I_{img} is visual input, I_{text} is textual input, the f_θ reconstructs x_0 from x_t, combining visual and textual input. More specifically, in our work, the x_0 is no longer an image but a layout annotation consisting of N bounding boxes. In inference, starting from random boxes, our model gradually modifies the position and size of boxes until a plausible layout is formed.

4.2 Feature Extractor

4.2.1 Image Encoder. Given a clean background image, we use ResNet-50 [7] with the Feature Pyramid Network (FPN) [17] to extract visual features. ResNet-50 has gained widespread popularity due to its exceptional performance in computer vision. Besides, we use FPN to produce multi-scale feature maps F, which consist of image features from low level to high level. Based on F, we extract RoI features [6] V with proposal x as follows:

$$V = RoIPooling(F, x), \tag{3}$$

where the shape of V is (C, W, H). In the training stage, the RoI feature comes from the real layout with Gaussian noise added, and it derives by random layout denoising in the inference stage.

4.2.2 Text Encoder. Given all the promotional slogans of the product on a poster, we extract textual features through a pre-trained language model RoBERTa [19]. We note that the product description is not simply repeating the product name, but highlighting the selling points of the product. For instance, if you want to promote a computer, you describe it as "high CPU performance" without mentioning "computer". Therefore, it is important to narrow the gap between the product description and the product itself. To address the problem, we gathered a vast product corpus of 200 million items from JD.com and adapt the same pretraining strategy which comprises Masked Language Model (MLM), Attribute-Value Prediction (AVP), and Tertiary Category Prediction (TCP) to fine-tune RoBERTa. For MLM, we randomly mask certain words from the input product title and feed it into the language model. This allows the model to predict the original sentence accurately. AVP and TCP are used to predict the value of a product based on its attribute and tertiary category. AVP is utilized to extract product values from the product description by utilizing product attribute queries. TCP involves the analysis and assessment of product information to determine the appropriate category. In order to let the model perceive the relationship between text length and layout, we supplement textual length embedding as a part of text features. Finally, we fuse the content features and length features of the text by concat operation, as the output of the text encoder, denoted as $L \in \mathbb{R}^{D_n \times d}$. It is worth noting that our method is not limited to Chinese. Migrating to another language only requires replacing the text encoder here.

Figure 5: The overview of the VTRAM. As illustrated in the figure, it takes as input text features, RoI features and corresponding coordinates. The coordinate information is first embedded into RoI features to get V_{ip}. Next, the scaled dot-product attention[29] is calculated using the visual position feature V_{ip} as the query, and text features L as the key and value.

4.3 Visual-Textual Relation-Aware Module

Instead of concatenating visual features and text features directly, we design a visual-textual relation-aware module to align the feature domain of the image and texts. The module is aware of the relationship between visual and textual elements and makes optimal use of features from both images and texts. This allows for a more comprehensive understanding of the content. In order to ensure a constant number of texts, we employ a method of padding additional vectors to reach a fixed number D_n. This approach offers the advantage of allowing our model to process texts of varying lengths.

Fig. 5 depicts the pipeline of VTRAM, which performs the multi-modal fusion of each RoI features $V_i \in \mathbb{R}^{C \times W \times H}$ and linguistic features $L \in \mathbb{R}^{D_n \times d}$ in two steps. First, to add explicit position information in visual features, the RoI feature V_i and its corresponding position embedding are concatenated to get the visual position feature V_{ip}:

$$V_{ip} = V_i \bigoplus P_g(G_i), \tag{4}$$

where the P_g is the project function, G_i is the coordinate of the i-th RoI.

Second, we use visual position feature V_{ip} as the query and linguistic feature maps L as the key and value:

$$\begin{aligned} V_{iq} &= P_q(V_{ip}), \\ L_k &= P_k(L), \\ L_v &= P_v(L), \end{aligned} \tag{5}$$

where the P_q, P_k, P_v are the 1×1 convolution function to convert the vectors into proper shape.

We calculate the final multi-modal feature M_i as follows:

$$M_i = P_o(softmax(\frac{V_{iq}^T L_k}{\sqrt{C}})L_v^T), \tag{6}$$

where the P_o is also a 1×1 convolution function. The multi-modal feature M_i gathers textual information that is closely related to RoI features, making visual features textual-aware.

4.4 Geometry Relation-Aware Module

We construct RoI features combining the results of the denoising process and image features, but these features of RoI are independent. To strengthen the position-aware relationship between RoI features, we designed Geometry Relation-Aware Module (GRAM) to allow the model to better learn the content information relationship between graph elements. The details are as follows. Firstly, given N RoIs, the relative position feature R_{ij} of two boxes l_i and l_j ($i, j \in \{1, 2, \ldots, N\}$) is calculated as :

$$R_{ij} = [\log(\frac{|x_i - x_j|}{w_j}), \log(\frac{|y_i - y_j|}{h_j}), \log(\frac{w_i}{w_j}), \log(\frac{h_i}{h_j})]. \tag{7}$$

Then, the 4-dimensional vectors are embedded to geometry weights by sin-cos encoding method [29] as R_{pij}.

$$\begin{aligned} PE_{(pos,2k)} &= \sin(\frac{pos}{10000^{8k/d_h}}), \\ PE_{(pos,2k+1)} &= \cos(\frac{pos}{10000^{8k/d_h}}), \\ R_p &= PE(R), \end{aligned} \tag{8}$$

Figure 6: The overview of GRAM. It exploits the relative positional relationships between elements. The input consists of two parts: relative position features R and RoI features V.

where the pos is the position and k is the dimension. The d_h we set in our experiment is 64. Finally, the geometry weights are normalized by the softmax function which prunes the weak pairwise relation and focuses more on the strong ones.

$$W = Softmax(R_p). \qquad (9)$$

What we need to emphasize is that there are different positioning strategies for different types of elements. The underlay should cover others while the rest elements should avoid overlapping. Therefore, we use extracted RoI features as element category information. To merge the position and category information, the extracted visual features V are flattened and transformed to vectors in d_t dimension by project function P. Finally, the visual embeddings multiply the geometry weights to get the final geometry features T:

$$T = W \cdot P((V')), \qquad (10)$$

where V' is the flattened form of V.

4.5 Layout Decoder

Similar to the task of object detection, the layout decoder predicts the category and coordinates of elements based on various types of RoI features. We construct the whole input of the layout decoder by fusing the outputs of VTRAM and GRAM, as well as the RoI features. The above process can be expressed as follows:

$$I_{decoder} = M \bigoplus T \bigoplus V, \qquad (11)$$

where $I_{decoder}$ represents the input of layout decoder, M is the output of VTRAM, T is the output of GRAM and V refers to the RoI features. $\bigoplus$ represents the fusion method of features, the concat

fusion used here. Then, these fused features are sent to the detection heads of bounding box regression and category prediction respectively to get the final coordinates and categories. Based on the above detection head results, we use box regression and classification losses to narrow the gap between the model's predictions and the ground truth, respectively. Meanwhile, in order to avoid excessive overlap between predicted boxes, we supplement giou loss as a penalty. The final weighted loss function is composed as follows:

$$Loss = \alpha_{cls} * L_{cls} + \alpha_{L1} * L_{L1} + \alpha_{giou} * L_{giou}, \qquad (12)$$

where L_{cls}, L_{L1} and L_{giou} respectively adopt focal loss [18], L1 loss and generalized IoU loss [25]. α_{cls}, α_{L1} and α_{giou} are weight coefficients for three different types of losses, which are set to 5, 5, and 1 respectively in this paper.

5 EXPERIMENT

In this section, we will compare the performance of our method and the SOTA method from both qualitative and quantitative perspectives.

5.1 Implementation Details

We implement the proposed method using Pytorch [23] and set the maximum diffusion step for sampling and denoising to 1000. Our model is trained using the AdamW [20] optimizer with the initial learning rate as 2.5×10^{-5} and the weight decay as 10^{-4}. We train the model for 100 epochs with batch size 16 on NVIDIA P40 GPU and the image size is normalized to 384×600 in order to improve training efficiency.

5.2 Evaluation Metrics

We follow the evaluation metrics in CGL-GAN [34], including three aspects: user study, composition-relevant measures and graphic measures.

For the user study, we randomly select 60 images from the test set and obtain the layout results corresponding to different methods and invite two groups of designers (five professional, twenty novice designers). Every designer needs to judge whether the layout result is qualified and select the best layout result for the same image. We denote the percentage passing the quality standard as P_{qs} and the percentage that hits the best layout as P_{best} (P_{qs}^* and P_{best}^* for the professional group) for each method.

Composition-relevant measures such as *Readability and visual balance R_{com}* and *Presentation of subjects* (R_{csub} and R_{shm}) are introduced in [34]. Readability and visual balance mean that when designing posters, designers tend to place text without underlays in a relatively flat area. R_{sub} and R_{shm} can reflect the degree of occlusion of key subjects, the lower the better. R_{occ} means the ratio of non-empty layouts predicted by models.

Graphic measures use the same indicators as in [34], such as alignment R_{ali}, overlap R_{ove} and R_{und}. R_{ove} excludes underlays and embellishments, because these two elements are generally attached to other types of elements. At the same time, redefine R_{und} to evaluate the influence of substrate elements on the layout quality. R_{und} and layout quality show a positive correlation.

Figure 7: Qualitative comparison results with SOTA methods. Each column layout represents the results obtained by different methods for the same image, and each row represents the layout results of the same method for different images.

Table 1: Comparison with content-aware methods.

Model	User study				Composition-relevant measures				Graphic measures		
	P_{qs}^* ↑	P_{best}^* ↑	P_{qs} ↑	P_{best} ↑	P_{shm} ↓	P_{com} ↓	P_{sub} ↓	P_{occ} ↑	P_{ali} ↓	P_{ove} ↓	P_{und} ↑
ContentGAN	26.1%	12.8%	30.6%	7.2%	23.610	31.930	0.767	**1.000**	0.009	0.065	0.840
CGL-GAN	28.3%	16.1%	44.4%	8.9%	21.670	16.040	0.772	0.875	**0.007**	0.081	0.732
Ours	**75.6%**	**66.7%**	**86.7%**	**78.9%**	**15.970**	**10.260**	**0.742**	0.997	0.008	**0.046**	**0.983**

5.3 Comparison with Content-Aware Methods

As mentioned in the previous chapters, ContentGAN and CGL-GAN are two generators considering the influence of image content on layout, so here is our main comparison model. We re-implement ContentGAN based on the released codes[1], and specifically add content feature extraction and text feature extraction modules consistent with our method. Meanwhile, we tried our best to re-implement the CGL-GAN method based on the details in the paper. The quantitative comparison results of the three methods are shown in Tab. 1. No matter whether in user study or composition-relevant metric, our method is obviously winning, which shows that the proposed

method has a better ability to represent the relationship between image content and layout.

The qualitative evaluation results of different models are shown in Fig. 7. The three columns on the left show that our model has a stronger subject representation ability, which can effectively highlight the subjects in posters such as commodities and models compared with other methods. From the results in the middle part, due to the introduction of the Visual-Textual Relation-Aware Module (VTRAM), the model can learn where the text should be placed to ensure the text readability and visual balance of the poster layout. The right part shows that our model can also strongly express the relationship between graph elements under the premise of ensuring that the products are not occluded.

---

[1]https://xtqiao.com/projects/content aware layout

Figure 8: Layout results with different amounts of text. The second to fourth columns represent a range of 1 to 3 input texts, respectively.

5.4 Comparison with Content-Agnostic Methods

Similarly, we also compare our model performance with recent content-agnostic SOTA methods [10, 13]. Based on the released code[2][3], we re-implement the above methods. As shown in Tab. 2, our model has great advantages in user study and composition-relevant because of the modeling relationship between image content and layout. But is less effective on graphic metrics. We attribute this to the fact that our model needs to consider image content information when generating layouts, such as considering visual balance factors or avoiding the main product area, etc. For the R_{und}, although our model does not exceed BLT, it is better than LayoutDM. Because of the introduction of the GRAM, the model learns the relationship between Underlay and other types of layout elements. As shown in the right part in Fig. 7, our model is more harmonious in the collocation of text and substrate.

5.5 Controllable Layout Generation

Our model can achieve controllable layout generation, which is also a highlight of our method. We show the layout results of the model under different constraints, which are (1) Text number and content; (2) Given partial layout.

Text number and content. As shown in Fig.8, the last three columns represent the layout results of the same background image under different text number constraints. Interestingly, we find that the number of text elements in the layout result is consistent with the number of input text, which proves that our model has learned the relationship between the number of texts and layout elements.

---

[2]https://github.com/CyberAgentAILab/layout-dm
[3]https://shawnkx.github.io/blt

Figure 9: Layout results with different text lengths (left column) and contents (right column).

Figure 10: Layout results under different user constraints.

As shown in Fig. 9, the left column indicates that given different text lengths, our method can generate boxes in the appropriate proportion, the right column represents the position of the element affected by the text content. It proves that the proposed model has a sufficient expression between literal semantic information and layout output.

Given partial layout. In order to verify whether the output results of the model are acceptable given the part layout, we conduct different experiments and the results are shown in Fig. 10. Our model can give qualified results, especially in the results of the third column, our model will not generate additional layouts without enough layout space, which shows that the model has strong constraints and generalization ability.

Table 2: Comparison with content-agnostic methods.

Model	User study				Composition-relevant measures				Graphic measures		
	$P_{qs}^* \uparrow$	$P_{best}^* \uparrow$	$P_{qs} \uparrow$	$P_{best} \uparrow$	$P_{shm} \downarrow$	$P_{com} \downarrow$	$P_{sub} \downarrow$	$P_{occ} \uparrow$	$P_{ali} \downarrow$	$P_{ove} \downarrow$	$P_{und} \uparrow$
BLT	57.2%	21.6%	57.8%	26.1%	22.450	28.540	0.765	1.000	**0.004**	**0.002**	**0.993**
LayoutDM	32.8%	13.8%	37.2%	22.8%	21.300	34.310	0.763	**1.000**	0.006	0.039	0.896
Ours	**75.6%**	**58.9%**	**82.2%**	**46.7%**	**15.970**	**10.260**	**0.742**	0.997	0.008	0.046	0.983

Table 3: Ablation studies of VTRAM. Ours* means our model without VTRAM.

Model	$P_{shm} \downarrow$	$P_{com} \downarrow$	$P_{sub} \downarrow$	$P_{occ} \uparrow$	$P_{ali} \downarrow$	$P_{ove} \downarrow$	$P_{und} \uparrow$
Ours*	17.450	12.720	0.764	0.989	0.010	0.053	**0.987**
Ours	**15.970**	**10.260**	**0.742**	**0.997**	**0.008**	**0.046**	0.983

Table 4: Ablation studies of GRAM. Ours* means our model without GRAM.

Model	$P_{shm} \downarrow$	$P_{com} \downarrow$	$P_{sub} \downarrow$	$P_{occ} \uparrow$	$P_{ali} \downarrow$	$P_{ove} \downarrow$	$P_{und} \uparrow$
Ours*	17.190	**10.120**	0.753	0.922	0.012	0.083	0.976
Ours	**15.970**	10.260	**0.742**	**0.997**	**0.008**	**0.046**	**0.983**

5.6 Ablation Studies

We conduct comparative experiments in the visual-textual relation-aware module, geometry relation-aware module, as well as the layout diversity and rationality.

Visual-Textual Relation-Aware Module. In order to verify the influence of visual and text attention features on the layout effect, we conduct ablation experiments. Specifically, we train two versions of the model on the same training data: (a) the model contains all modules; (b) the model removes VTRAM. The results can be seen in Tab. 3. Due to the introduction of the text and image attention mechanism, the model has learned content information related to the composition of the image, which greatly improves the composition-relevant metrics without sacrificing the effectiveness of graph metrics to a certain extent. We believe that multi-modal deep semantic features have a more accurate expression for layout elements.

Geometry Relation-Aware Module. Geometry Relation-Aware Module (GRAM) is to obtain more robust and accurate box coordinates and sizes after the diffusion process. We remove the GRAM from the proposed model as a ablation comparison model. As shown in Tab. 4, the model with GRAM has a 0.4% reduction on R_{ali}, a 0.07% improvement on R_{und} and a 3.7% reduction on R_{ove}, which is attributed to the more accurate description of the boxes in the process of generating the layout. In particular, the performance of composition-relevant metrics has also been improved, because the influence of image information on the position of elements is also considered in the introduction of GRAM. In general, GRAM can achieve a balance in the improvement of composition-relevant metrics and graphic metrics.

Layout diversity and rationality. Because our method will give some random layout boxes at the beginning of the inference stage, in order to evaluate the layout diversity and rationality of the

Figure 11: Generated layouts under different random seeds. Each row is the result of the same input image under different random seeds, and each column the different images under the same random seed.

model, we give qualitative experimental results. From left to right, Fig. 11 shows the layout results corresponding to five different layouts by random seeds at the beginning of inference. From top to bottom, Fig. 11 also shows the layout results of different images under the same random seed. Although the resulting layout results are different, they are all reasonable, indicating the diversity and rationality of the layout model.

6 CONCLUSION

In this paper, we propose a relation-aware diffusion model to generate poster layouts, in which the relationship between visual and textual contents and the relationship between elements are considered to help get pleasant layouts. To better integrate visual and textual features, we design a Visual-Textual Relation-Aware Module (VTRAM) to learn the relationship between visual and textual contents. As the coordination of element positions is important for layout, a Geometry Relation-Aware Module (GRAM) is employed to enhance features based on the relative position between elements. In addition, we build a large poster layout dataset, named CGL-Dataset V2. We conduct extensive experiments to prove that the proposed method significantly outperforms the existing methods and can achieve controllable generation. Ablation studies also demonstrate the effectiveness of VTRAM and GRAM.

REFERENCES

[1] Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. 2021. Structured denoising diffusion models in discrete state-spaces. In *Advances in Neural Information Processing Systems*, Vol. 34. 17981–17993.

[2] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. 2023. Align Your Latents: High-Resolution Video Synthesis With Latent Diffusion Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 22563–22575.

[3] Ying Cao, Antoni B Chan, and Rynson WH Lau. 2012. Automatic stylistic manga layout. *ACM Transactions on Graphics (TOG)* 31, 6 (2012), 1–10.

[4] Yunning Cao, Ye Ma, Min Zhou, Chuanbin Liu, Hongtao Xie, Tiezheng Ge, and Yuning Jiang. 2022. Geometry Aligned Variational Transformer for Image-conditioned Layout Generation. In *Proceedings of the 30th ACM International Conference on Multimedia*.

[5] Shoufa Chen, Peize Sun, Yibing Song, and Ping Luo. 2022. DiffusionDet: Diffusion Model for Object Detection. *arXiv preprint arXiv:2211.09788* (2022).

[6] Ross Girshick. 2015. Fast R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[8] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising Diffusion Probabilistic Models. In *Advances in Neural Information Processing Systems*, Vol. 33. 6840–6851.

[9] Mude Hui, Zhizheng Zhang, Xiaoyi Zhang, Wenxuan Xie, Yuwang Wang, and Yan Lu. 2023. Unifying Layout Generation with a Decoupled Diffusion Model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 1942–1951.

[10] Naoto Inoue, Kotaro Kikuchi, Edgar Simo-Serra, Mayu Otani, and Kota Yamaguchi. 2023. LayoutDM: Discrete Diffusion Model for Controllable Layout Generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 10167–10176.

[11] Charles Jacobs, Wilmot Li, Evan Schrier, David Bargeron, and David Salesin. 2003. Adaptive grid-based document layout. *ACM transactions on graphics (TOG)* 22, 3 (2003), 838–847.

[12] Akash Abdu Jyothi, Thibaut Durand, Jiawei He, Leonid Sigal, and Greg Mori. 2019. LayoutVAE: Stochastic Scene Layout Generation From a Label Set. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

[13] Xiang Kong, Lu Jiang, Huiwen Chang, Han Zhang, Yuan Hao, Haifeng Gong, and Irfan Essa. 2022. BLT: Bidirectional Layout Transformer For Controllable Layout Generation. In *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XVII*. 474–490.

[14] Ranjitha Kumar, Jerry O. Talton, Salman Ahmad, and Scott R. Klemmer. 2011. Bricolage: Example-Based Retargeting for Web Design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, 2197–2206.

[15] Hsin-Ying Lee, Lu Jiang, Irfan Essa, Phuong B Le, Haifeng Gong, Ming-Hsuan Yang, and Weilong Yang. 2020. Neural design network: Graphic layout generation with constraints. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*. 491–506.

[16] Jianan Li, Jimei Yang, Aaron Hertzmann, Jianming Zhang, and Tingfa Xu. 2019. LayoutGAN: Generating Graphic Layouts with Wireframe Discriminators. In *International Conference on Learning Representations*.

[17] Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. 2017. Feature Pyramid Networks for Object Detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[18] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. 2017. Focal Loss for Dense Object Detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

[19] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *ArXiv* abs/1907.11692 (2019).

[20] Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*.

[21] Peter O'Donovan, Aseem Agarwala, and Aaron Hertzmann. 2014. Learning Layouts for Single-PageGraphic Designs. *IEEE Transactions on Visualization and Computer Graphics* 20 (2014), 1200–1213.

[22] X. Pang, Ying Cao, Rynson W. H. Lau, and Antoni B. Chan. 2016. Directing user attention via visual flow on web designs. *ACM Transactions on Graphics (TOG)* 35 (2016), 1 – 11.

[23] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, Vol. 32.

[24] Chunyao Qian, Shizhao Sun, Weiwei Cui, Jian-Guang Lou, Haidong Zhang, and Dongmei Zhang. 2020. Retrieve-then-adapt: Example-based automatic generation for proportion-related infographics. *IEEE Transactions on Visualization and Computer Graphics* 27, 2 (2020), 443–452.

[25] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. 2019. Generalized Intersection Over Union: A Metric and a Loss for Bounding Box Regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[26] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. 2023. 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 (CVPR)*. 22500–22510.

[27] Jiaming Song, Chenlin Meng, and Stefano Ermon. 2020. Denoising Diffusion Implicit Models. *ArXiv* abs/2010.02502 (2020).

[28] Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. 2022. Resolution-Robust Large Mask Inpainting With Fourier Convolutions. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 2149–2159.

[29] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, Vol. 30.

[30] Cheng-Fu Yang, Wan-Cyuan Fan, Fu-En Yang, and Yu-Chiang Frank Wang. 2021. LayoutTransformer: Scene Layout Generation With Conceptual and Spatial Diversity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 3732–3741.

[31] Lap-Fai Yu, Sai-Kit Yeung, Chi-Keung Tang, Demetri Terzopoulos, Tony F Chan, and Stanley J Osher. 2011. Make it home: automatic optimization of furniture arrangement. *ACM Transactions on Graphics (TOG)* 30, 4 (2011), 1–12.

[32] Yuxin Zhang, Nisha Huang, Fan Tang, Haibin Huang, Chongyang Ma, Weiming Dong, and Changsheng Xu. 2023. Inversion-Based Style Transfer With Diffusion Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 10146–10156.

[33] Xinru Zheng, Xiaotian Qiao, Ying Cao, and Rynson WH Lau. 2019. Content-aware generative modeling of graphic design layouts. *ACM Transactions on Graphics (TOG)* 38, 4 (2019), 1–15.

[34] Min Zhou, Chenchen Xu, Ye Ma, Tiezheng Ge, Yuning Jiang, and Weiwei Xu. 2022. Composition-aware Graphic Layout GAN for Visual-Textual Presentation Designs. In *IJCAI*. 4995–5001.