

# AutoPoster: A Highly Automatic and Content-aware Design System for Advertising Poster Generation

Jinpeng Lin\*  
Min Zhou\*  
Alibaba Group  
Beijing, China  
jplinforever@gmail.com  
yunqi.zm@alibaba-inc.com

Ye Ma  
Alibaba Group  
Beijing, China  
maye.my@alibaba-inc.com

Yifan Gao  
University of Science and Technology  
of China  
Hefei, China  
eafn@mail.ustc.edu.cn

Chenxi Fei  
Alibaba Group  
Hangzhou, China  
corey.fcx@alibaba-inc.com

Yangjian Chen  
Zhang Yu  
Alibaba Group  
Hangzhou, China  
yicai.cyj@alibaba-inc.com  
zy99945@alibaba-inc.com

Tiezheng Ge<sup>†</sup>  
Alibaba Group  
Beijing, China  
tiezheng.gt@alibaba-inc.com

## ABSTRACT

Advertising posters, a form of information presentation, combine visual and linguistic modalities. Creating a poster involves multiple steps and necessitates design experience and creativity. This paper introduces AutoPoster, a highly automatic and content-aware system for generating advertising posters. With only product images and titles as inputs, AutoPoster can automatically produce posters of varying sizes through four key stages: image cleaning and retargeting, layout generation, tagline generation, and style attribute prediction. To ensure visual harmony of posters, two content-aware models are incorporated for layout and tagline generation. Moreover, we propose a novel multi-task Style Attribute Predictor (SAP) to jointly predict visual style attributes. Meanwhile, to our knowledge, we propose the first poster generation dataset that includes visual attribute annotations for over 76k posters. Qualitative and quantitative outcomes from user studies and experiments substantiate the efficacy of our system and the aesthetic superiority of the generated posters compared to other poster generation methods.

## CCS CONCEPTS

• **Information systems** → **Multimedia content creation**; • **Computing methodologies** → **Image-based rendering**; **Graphics systems and interfaces**.

\*Both authors contributed equally to this research.

<sup>†</sup>Corresponding author

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

MM '23, October 29–November 3, 2023, Ottawa, ON, Canada

© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 979-8-4007-0108-5/23/10...\$15.00

<https://doi.org/10.1145/3581783.3611930>

## KEYWORDS

advertising poster generation, automatic system, design tool

### ACM Reference Format:

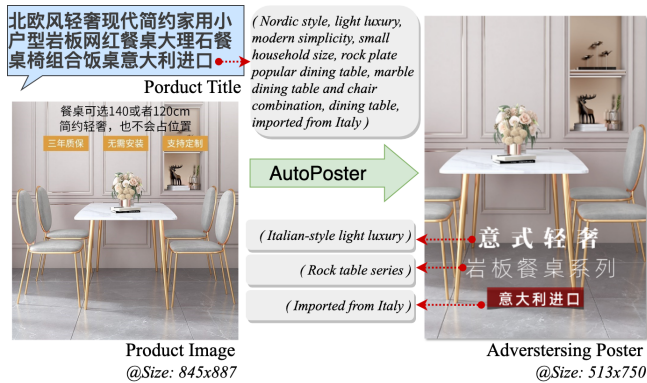
Jinpeng Lin, Min Zhou, Ye Ma, Yifan Gao, Chenxi Fei, Yangjian Chen, Zhang Yu, and Tiezheng Ge. 2023. AutoPoster: A Highly Automatic and Content-aware Design System for Advertising Poster Generation. In *Proceedings of the 31st ACM International Conference on Multimedia (MM '23)*, October 29–November 3, 2023, Ottawa, ON, Canada. ACM, New York, NY, USA, 11 pages. <https://doi.org/10.1145/3581783.3611930>

## 1 INTRODUCTION

Conveying comprehensive product information is crucial for e-commerce platforms, especially when potential customers are browsing without a clear purchase objective. Advertising posters, which incorporate product images and diverse graphic design elements such as taglines, underlays, and logos, effectively display product appearances, brands, and unique selling points.

With multiple elements of different modalities to consider, poster creation can be a challenging task that involves three main aspects and requirements. First, the layout of images and various graphic elements must meet the target size requirements and be harmonious, while highlighting the main subject. Second, concise and structured taglines must be generated based on the product information to quickly convey messages to customers. Third, to achieve a harmonious and visually appealing poster, all elements within the poster are interconnected. Therefore, manual poster creation often requires a significant amount of time and creativity, particularly when creating posters of different sizes for various products. While some automated methods [9, 24, 27, 33] are available to assist with poster creation, they often lack sufficient automation and require users to prepare target-sized images and tagline content in advance. Additionally, they depend heavily on manual rules to ensure visual effects, resulting in reduced flexibility and diversity.

In this paper, we propose a novel and highly automated method for generating advertising posters, called AutoPoster. As shown in Fig. 1, an arbitrary product image and product description are all the information provided by the user to generate a complete poster



**Figure 1: Given a product image, product title, and target poster size, our AutoPoster method can automatically generate the corresponding poster. To enhance readability, English translations of the Chinese taglines are *italicized*, and subsequent figures are treated in the same manner.**

of a specified size, and the user can continue to adjust the layout, tagline content and style attributes of the graphic elements. Professional designers often first organize visual elements (e.g. product image) and then prepare graphic elements (e.g. taglines, logos) accordingly. Besides, to achieve a harmonious and visually appealing poster, all elements within the poster are interconnected. Guided by these principles, AutoPoster can be divided into four steps: 1) image cleaning and retargeting. Assisted with detection, inpainting, saliency, and outpainting models, graphic elements on the product image are erased, and the image is retargeted to the target poster size while maintaining the subject unchanged. 2) layout generation. We utilize CGL-GAN [37] and ICVT [2] to arrange the number and position of graphic elements (text, logo, and underlay) according to image contents. 3) tagline generation. A multi-modal generative model [7] is used to yield tagline content based on image content, product information, and layouts. 4) style attribute prediction. We introduce an innovative Style Attribute Predictor (SAP), a non-autoregressive and multi-task transformer that models the visual relationship between images and graphic elements, as well as that within graphic elements. In other words, SAP can simultaneously predict the font typography, dominant color, stroke color, and gradient color of taglines and underlays according to image contents and graphic layouts.

We trained several models using a dataset of 76,960 annotated posters, which is the first large-scale dataset for poster generation. The annotation encompasses tagline content, graphic element position, and style attributes. To annotate font typography that is difficult for humans, we trained a font recognition model with self-supervised learning. Qualitative and quantitative experiments show the effectiveness and superiority of AutoPoster and SAP.

The contributions of this paper can be summarized as follows:

- We propose a novel approach to model poster design through four key steps: image cleaning and retargeting, layout generation, tagline generation, and style attribute prediction. Our method requires only product images, information such as

product titles, and desired poster dimensions from users to produce complete posters.

- For style attribute prediction, we introduce a novel multi-task and non-autoregressive multi-modal sequence model that learns the relationships among graphic element attributes, their positions, and image content.
- To highlight and maintain subjects in poster design, we utilize multiple models to clean product images and retarget them to the target sizes.
- To our knowledge, we constructed the first large poster generation dataset, which contains annotations of tagline content and various visual attributes of graphic elements. And we train a self-supervised model to label the font typography.

## 2 RELATED WORK

The development of an information representation system encompasses various interdisciplinary tasks and topics. We categorize the related work from the comprehensive automatic design generation system down to the individual phases.

**Automatic information presentation generation system.** To save manual effort, various automatic methods [15, 19, 30, 34] for information presentation have been developed. Two similar works are Vempati *et al.* [26] and Vinci [9], which can automatically generate posters based on product images and taglines. AutoPoster differs in three main ways: a) it can generate more flexible layouts and any number of taglines on the image based on the product title; b) it can handle a wider range of product images, including images with texts and photo-style images (naturally photographed product images); c) it is equipped with an image-aware style attribute prediction model and can automatically predict more fine-grained attributes, such as font and stroke color. While Vinci and Vempati *et al.* [26] only adjust the text color by the contrast between texts and backgrounds, ensuring readability but ignoring color harmonization.

**Image retargeting.** Image retargeting means changing image sizes while keeping the visual content as much as possible. The most straightforward algorithm is cropping [26, 34]. However, when the aspect ratios of the product image size and target poster size are quite different, these methods will inevitably crop off parts of subjects, affecting the product presentation. Warping-based methods [1, 3] armed with deep learning can avoid this problem. However, they aim at retaining the whole visual content and may alter the product regions, which is also an unwanted situation in poster design. In contrast to these methods, we propose a combination of cropping and outpainting as a way to tackle the above problems.

**Layout generation.** For graphic layout generation, previous works [11, 17, 20] usually utilize templates or heuristic methods. Recently, CGL-GAN [37] and ICVT [2] utilize transformers [25] to generate image-content-aware layouts and only require posters with element positions labeled for training. We introduce these two methods to produce layouts conditioned on product images.

**Tagline generation.** Existing automatic poster generation systems [24, 26, 32] lack the ability to generate taglines, which users need to input manually. Generating taglines automatically on the image would simplify user interactions. The CapOnImage model [7]

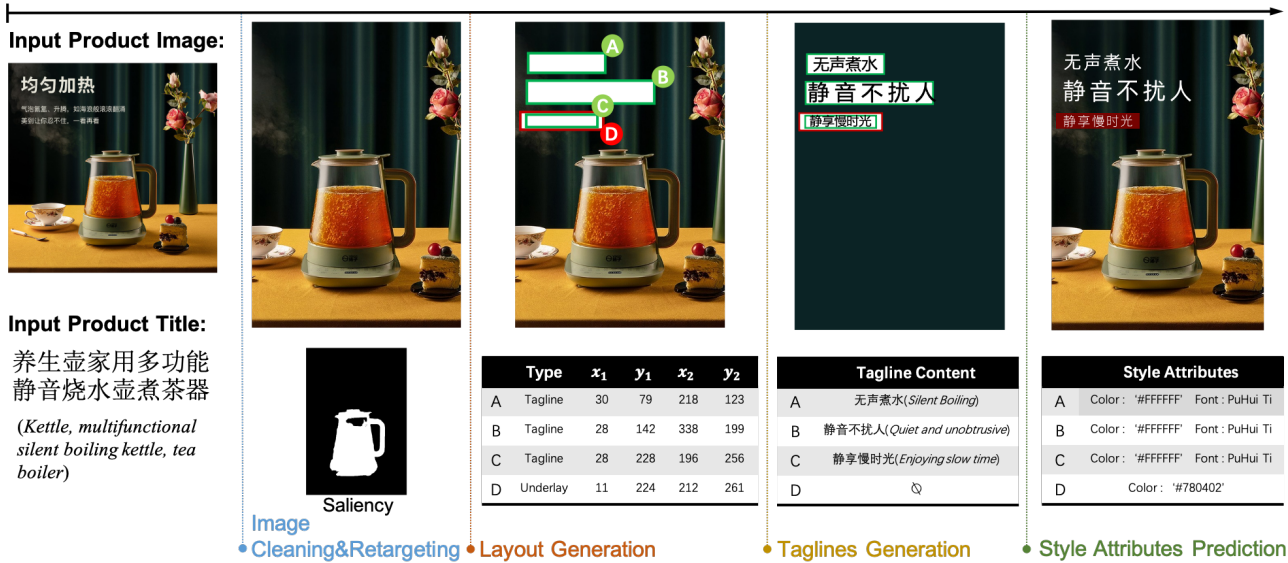


Figure 2: AutoPoster creates posters by following a four-stage process that utilizes product images and titles. The stages are displayed in a sequential manner from left to right.

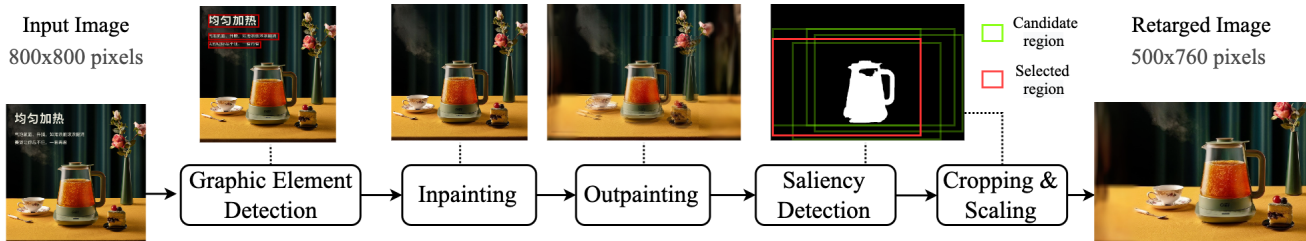


Figure 3: Five key steps for image cleaning and retargeting pipeline.

produces taglines for various positions on the product image based on visual and textual contextual information. In this work, we introduce the CapOnImage model for tagline generation.

**Style attribute prediction.** The style attributes in this paper can be divided into two main categories: font typography and color. For predicting font typographies, Zhao *et al.* [36] proposes a model to choose a font given a web design. FontMatcher [4] matches a harmonious font to the input image. Visual Font Pairing [14] focuses on finely recommending suitable paired fonts for a known font. The above methods pick fonts without considering both the image contexts and font selections of other elements. We design a transformer-based model and utilize cross-attention to perceive the image contexts and self-attention for internal element contexts. For color design, some works [12] use color harmonic models [23] for rule-based color selection. [32] adopts the topic-dependent templates in harmonic color design. Leveraging color theory [13], [24] extract the color palette of the whole image and determine the color considering the contrast ratio to ensure clear visibility. In contrast to these rule-based algorithms, we take a data-driven approach to color prediction similar to the image colorization problem [35] and extend the font prediction model to a multi-task one.

### 3 METHODOLOGY

In this section, we begin by defining the design space, and then we provide a detailed description of each module in AutoPoster.

#### 3.1 Formalized Poster Design Space

As mentioned earlier, the entire process is divided into four steps illustrated in Fig. 2. The image cleaning and retargeting module changes product images to the desired size, maintaining subjects and eliminating existing graphic elements. The layout generation module determines the quantity, variety, and position of graphic elements, while the tagline generation module produces pertinent captions. To guarantee a visually appealing design, the style predictor estimates various visual stylistic attributes.

Table 1: Attributes to be predicted for each type of element.

Element Type	Dominate Color	Gradient Color	Stroke Color	Font Typography
Underlay	✓	✓	-	-
Text	✓	✓	✓	✓

The corresponding design space can be structured in three dimensions: **Layout design space**. A layout is an arrangement of  $N$  graphic elements  $\{e_1, e_2, \dots, e_N\}$ . Each element comprises the

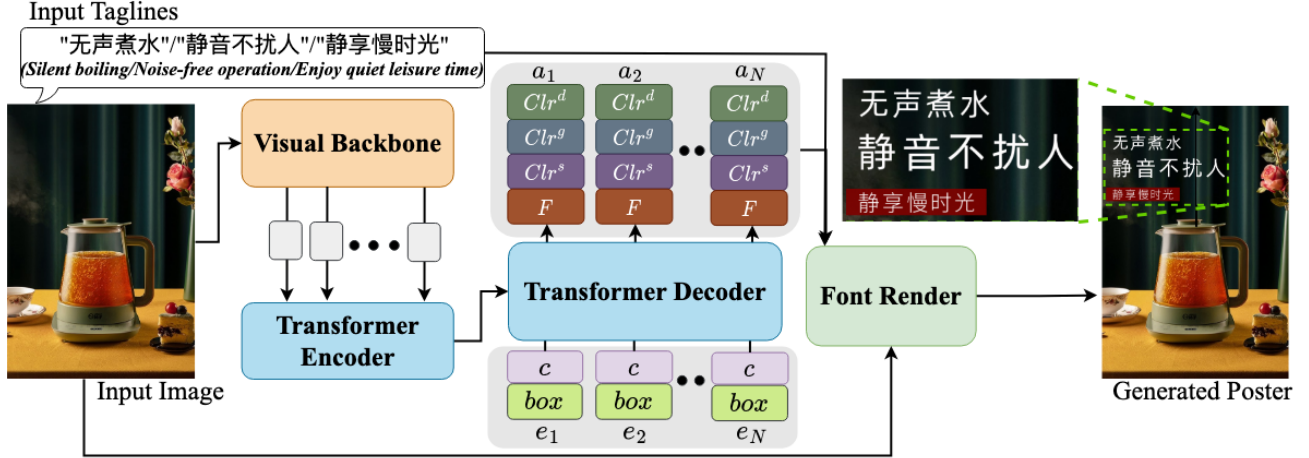


Figure 4: Model framework of the proposed SAP, which takes the image  $I$  and layout  $\{e_1, \dots, e_N\}$  as input, and outputs attributes  $a_i$  for each element  $e_i$ . With predicted attributes  $\{a_1, \dots, a_N\}$ , layout, taglines and  $I$ , an advertising poster can be rendered.

category, position, and size, represented as  $e_i = [c, x_1, y_1, x_2, y_2]$  ( $c \in \{\text{background image, logo, underline, tagline}\}$ ). **Tagline design space.** This paper focuses on creating Chinese taglines. A tagline of length  $L$  is denoted as  $[w_1, w_2, \dots, w_L]$ , where  $w_i$  represents a Chinese character. **Visual style attributes design space.** The style attributes of graphic elements here are dominant color  $Clr^d$ , stroke color  $Clr^s$ , gradient color  $Clr^g$ , and font typography  $F$ . For an element  $e_i$ , its visual style attributes are formulated as  $a_i = \{Clr_i^d, Clr_i^s, Clr_i^g, F_i\}$ . As shown in Table 1, depending on the category,  $e_i$  contains different subsets of  $a_i$ .

### 3.2 Image Cleaning and Retargeting

As mentioned in Section 1, there are some rules for the background image of advertising posters: product regions must remain unaltered, surrounding regions can be modified to a greater extent, and the entire image must be visually harmonious and clean. To meet these requirements, we've developed a five-step image cleaning and retargeting pipeline, which is illustrated in Fig. 3.

**Graphic element detection.** An object detection model [5] is trained to detect logos, taglines, and underlays in input images. The detected locations are used to create binary masks, indicating areas for inpainting. **Inpainting.** We employ an off-the-shelf inpainting model [22] to remove the detected graphic elements and ensure the image is clean for later design processes. **Outpainting.** We use a self-supervised approach to train an outpainting model [16]. This model is used to extend image regions seamlessly, avoiding cropping issues due to significant differences in aspect ratios between the target and image. **Saliency detection.** An off-the-shelf saliency detection model [29] is employed to detect product regions with saliency maps. **Cropping and scaling.** Utilizing saliency maps, we initially crop images to achieve the target aspect ratio before scaling them to the desired size. We slide the maximum sub-window with the target aspect ratio across the saliency map to find the region with the highest saliency scores to ensure the product is complete and prominent. We use integral image [28] and convolution techniques to speed up this process. Multiple optimal regions

may exist, so preferences are set based on the target aspect ratio to determine the final selected region. As Fig. 3 shows, when the target aspect ratio is greater than one, we find the candidate region with the product center closest to the left or right center of the region. Conversely, for images with aspect ratios less than one, we set the preference at the bottom center.

### 3.3 Layout Generation

After retargeting the product image to match the poster size, we arrange logos, underlays, and taglines using content-aware layout generation models: CGL-GAN [37] and ICVT [2]. These models utilize product images and saliency maps as input and transformer architecture to learn the relationship between graphic elements and product images. The output is the graphic element layout, denoted as  $\{e_1, e_2, \dots, e_N\}$ . The training process only requires labeled element positions in posters, not pairs of clean images and posters.

### 3.4 Tagline Generation

Once the layouts have been set, we generate taglines with various styles and content to enhance the appeal of posters. The tagline content may include selling points, click-through guides, benefit points, and more. We follow the approach of [7] and design an automatic tagline generation module. This module utilizes multi-modal text generation techniques to generate suitable tagline content by taking into account the product image, product title (description information), and layout of all graphic design elements.

### 3.5 Style Attribute Prediction

Rendering graphic elements with style attributes onto the background image is necessary for creating a complete poster, and the rendering quality significantly impacts the readability, information conveyance, and visual aesthetic of the poster. In this section, we present the SAP, which can estimate visual style attributes for all graphic elements based on the product image and layout.

**3.5.1 Discrete Representation of Attributes.** As shown in Table 1, the style attributes can be summarized as color and font. For

model training, we make a uniform discrete representation of these two attributes. Following [35], we perform color-related attributes (dominant color, stroke color, and gradient color) in CIE Lab color space, in which the color distance is more consistent with human perception. Specifically, we quantize the ab space into bins with grid size 10 and only get 313 values which are in-gamut like [35] and evenly quantize the lightness of the LAB color to 10 values. Therefore, each color can be represented by two discrete values. For the font attribute of taglines, we use 62 commonly used font typography for design, so  $F \in [1, 62]$ .

**3.5.2 Model Architecture.** Fig. 4 displays the model overview of the style attribute predictor, *SAP*. It is based on the transformer model[25] and contains an encoder and a decoder. *SAP* takes an image  $I$  and graphic layout information  $E = \{e_i = (x_{1i}, y_{1i}, x_{2i}, y_{2i}, c_i) | i \in [1, N]\}$  as input, and outputs the attributes of all graphic elements  $\{a_i = \{Clr_i^d, Clr_i^s, Clr_i^g, F_i\} | i \in [1, N]\}$ .

**Encoder.** A CNN visual backbone is first used for extracting local feature information from product images. More precisely, given an image  $I \in \mathbb{R}^{H \times W \times C}$  in RGB space. We first use a ResNet [10] to encode  $I$  into a  $P \times P$  visual feature with a downscale factor of 16. Then the 2D features are flattened with position embedding and sent to the ViT [6] transformer to capture long-distance relations. Finally, the encoder generates a visual feature  $f_v \in \mathbb{R}^{l \times d}$ , where  $l = HW/P^2$  is the length of the flattened feature sequence, and  $d$  is the embedding dimension.

$$f_v = \text{encoder}(I) \quad (1)$$

**Decoder.** The decoder takes the visual feature  $f_v$  and graphic layout information  $E$  as input. Each block of the decoder consists of multiple layers of self-attention and cross-attention. The self-attention layer models the relationship between different graphic elements, while the cross-attention mechanism queries visual memory features for graphic elements. The decoder predicts the attributes  $a_i$  for each element  $e_i$ , which can be formulated as:

$$\{a_1, \dots, a_N\} = \text{decoder}(f_v, \{e_1, \dots, e_N\}) \quad (2)$$

The attributes of each element can be decoded in a non-autoregressive [8] or autoregressive [21] manner. We analyze the differences in the subsequent experiments.

**3.5.3 Multi-Task Training.** We adopt a joint optimization strategy for inter-connected style attributes, namely  $Clr^d, Clr^s, Clr^g, F$ . For instance, the selection of dominant color can impact that of stroke color. Moreover, designers may add a stroke to the tagline if the dominant color is insufficient for readability. Thus, we allow *SAP* to predict multiple attributes concurrently. Our experiments indicate that multi-task joint optimization surpasses single-task learning, thereby revealing the fundamental interconnectedness of these attributes. As style attributes are often long-tail distributed, we utilize focal-loss [18] to optimize the model. Thus, *SAP* is trained with the following loss:

$$\{\hat{a}_1, \dots, \hat{a}_N\} = \text{SAP}(I, E) \quad (3)$$

$$\text{Loss} = \frac{1}{NK} \sum_{i=1}^N \sum_{j=1}^K \lambda_j \text{FocalLoss}(\hat{a}_i^j, a_i^j) \quad (4)$$

where  $\lambda_j$  is used to adjust the loss weight of individual attributes,  $K$  denotes the number of attributes, and  $\hat{a}_i^j$  represents the predicted  $j$ -th attribute of the  $i$ -th element.

**3.5.4 Underlay Retrieval.** To increase the diversity of underlay shapes, we generated a database of various underlays extracted from SVGs designed by professionals. Each underlay is matched with the nearest neighbor from the database according to its size and ratio. The selected underlay is adjusted to fit the predicted position of the underlay element and its color attributes are altered to match the prediction of our *SAP*.

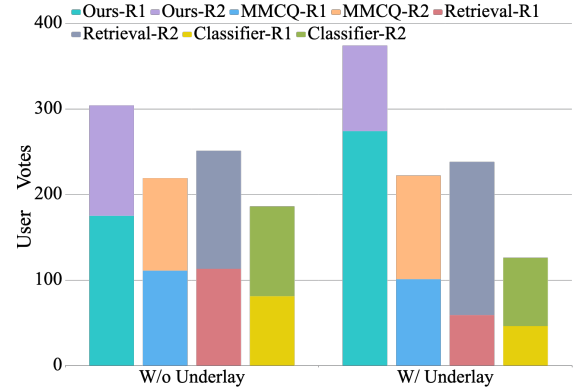


Figure 5: User survey results of ours and other methods on the R1 and R2 scores.

## 4 EXPERIMENT

In this section, we evaluate our proposed method using real-world data. In this section, we evaluate our proposed method using real-world data. We start by introducing the dataset and then demonstrate the effectiveness of *SAP* through comparisons with other methods and ablation studies. Next, we compare the overall performance of our proposed method with other poster generation approaches and show its ability to generate posters of various sizes.

### 4.1 Dataset

We obtained 76,960 advertising posters from an e-commerce platform and used 69,249 for training and 7,711 for quantitative tests. Additionally, we collected 46 product images for a user study. The posters are manually designed, providing a diverse set of layouts, taglines, and visual styles across various product categories. We annotated each poster for the layout, tagline content, and color attributes of each graphic design element. For tagline font typography, we trained a font-weight-aware font classifier using a self-supervised approach. Dataset is available at <https://tianchi.aliyun.com/dataset/159829>.

### 4.2 Compare SAP with Other Methods

We compare our *SAP* model to the following three methods for coloring taglines since there are few works focusing on this task.

- **MMCQ** [24]: The approach involves extracting the main colors from both local patch and global images separately. Afterward, the color with the highest contrast between extracted colors is selected as the predicted result.

**Table 2: The ablation study results for SAP. The best results are bolded, and the second-best results are underlined.**

Methods	D-ab (MSE↓)	D-light(MAE↓)	G(Acc↑)	G-ab(MSE↓)	G-light(MAE↓)	S(Acc↑)	S-ab(MSE↓)	S-light(MAE↓)
<i>Random</i>	62.71	3.90	5.0%	86	2.89	16.3%	86.3	4.48
<i>D-color</i>	22.81	<b>2.28</b>	-	-	-	-	-	-
<i>D-color + S-color</i>	22.60	3.07	-	-	-	<u>35.2 %</u>	<b>28.24</b>	<u>2.99</u>
<i>D-color + G-color</i>	<u>22.52</u>	2.43	27.2%	<u>41.50</u>	<u>1.78</u>	-	-	-
ours w/AR	25.64	3.0	<u>29.0%</u>	46.01	1.90	28.4%	30.37	3.23
ours	<b>22.19</b>	<u>2.33</u>	<b>30.0%</b>	<b>41.44</b>	<b>1.78</b>	<b>39.3%</b>	<u>28.40</u>	<b>2.76</b>

- **Retrieval:** This method matches tagline colors by using color histograms of tagline regions in images. During inference, it selects the most similar tagline from the library based on the histograms and uses its color as the result.
- **Classifier** [36]: The classifier network estimates the category of tagline color after quantifying the color space.

**4.2.1 Qualitative Evaluation.** We assess the efficacy of our SAP model via a user survey, which evaluates 30 groups of four posters produced by various methods concerning their readability, harmony, and visual aesthetics. Out of these, 15 groups require color prediction exclusively for taglines (w/o underlays), while the remaining 15 groups involve underlays (w/ underlays), for which our approach predicts the underlay color. Participants select the best (R1) and second-best images (R2) within each group. A higher R1 indicates better outcomes, and R1+R2 reflects greater robustness with fewer inferior cases. The evaluation comprises 30 participants, and the voting results are illustrated in Fig. 5. Our method has the most votes in both R1 and R1+R2, reflecting its superior visual quality. Furthermore, the prediction of underlay color enhances the vote percentage of our method.

Fig. 6 shows a visual comparison between SAP and other methods. As can be seen, the **Classifier** approach is prone to predict the color with the highest frequency, such as red, due to the problem of imbalanced color categories in the training data. The method of using **MMCQ** to extract the main colors of an image and increase the contrast of text color can lead to text that is unreadable. Moreover, none of the three competing methods take into account the correlation between colors for multiple sentences, resulting in inconsistent color combinations. In contrast, SAP can generate readable results with harmonious color combinations between different graphic elements. It can be seen that adding strokes and underlay further improves the readability of the tagline and the overall aesthetics.

**4.2.2 Quantitative Evaluation.** We assessed SAP, MMCQ, Retrieval, and Classifier on the test dataset using the D-ab & D-light metric. As shown in the Table 3, in consideration of the overall LAB space, SAP’s predicted color is the most accurate. It is worth noting that SAP can estimate the dominant color, stroke color, gradient color, and font together, while the three comparison methods can only estimate the dominant color.

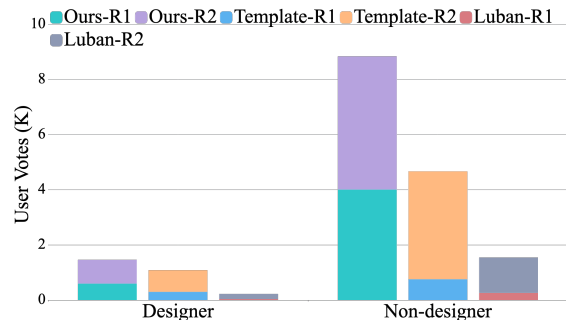
**Table 3: Quantitative comparison results of SAP.**

Metric	MMCQ	Classifier	Retrieval	SAP
D-ab↓	<b>22.08</b>	40.72	33.40	<u>22.19</u>
D-light↓	4.76	2.87	<u>2.65</u>	<b>2.33</b>

**Figure 6: Our SAP compared with Classifier [36], MMCQ [24], and Retrieval method for color prediction.**

### 4.3 Ablation Studies

We make two ablation studies to analyze the performance of SAP and gain insights into poster image design. Three metrics are used: a) dominant, gradient, and stroke color prediction error in the AB dimension of the LAB color space (D-ab, G-ab, and S-ab, respectively, calculated using mean squared error); b) dominant, gradient, and stroke color prediction error in the lightness dimension of the LAB color space (D-light, G-light, and S-light, respectively, calculated using mean absolute error); c) prediction accuracy of whether design elements possess gradient or stroke (G/S).

**Figure 7: Overall user survey results on R1 and R2 votes for different methods.**

**Single-task vs. multi-task learning.** To verify the effectiveness of joint optimization, we conduct four comparative experiments: 1) *Random*; 2) *D-color*: a SAP model only predicting the



Figure 8: Qualitative comparison of posters generated by various methods: (a) User-input Chinese product titles (descriptions); (b) User-uploaded images; (c, d, e) Posters produced by each of the three methods. "Unclean image" means there are no graphic design elements on the original input image, and vice versa.

dominant color; 3) *D-color + S-color*: a SAP model predicting both the dominant and stroke color; 4) *D-color + G-color*: a SAP model predicting the dominant and gradient color. Ours simultaneously

predicts all three color attributes (dominant, stroke, and gradient color). The results are presented in Table 2. As the number of prediction tasks (attributes) increases, the performance improves. This



**Figure 9: Generated posters by AutoPoster in arbitrary sizes. The input product images are highlighted with red box.**

suggests a correlation between the different color attributes in design, and learning one task can enhance the learning of other tasks.

**Autoregressive vs. non-autoregressive manner.** When designing the poster, the designer can follow two approaches: a) Decide on the overall layout first and then choose the visual style, considering the relationship between elements. b) Set the visual attributes of the most important element first and design the rest accordingly. These lead to two decoding methods: *NAR* (non-autoregressive) and *AR* (autoregressive). For a given product image  $I$  and layout  $E$ , the probability of predicted attributes can be determined as follows:

$$\begin{aligned}
 \text{NAR} : P_{\theta}(a_1, \dots, a_N | I, E) &= \prod_{t=1}^N P_{\theta}(a_t | I, E) \\
 \text{AR} : P_{\theta}(a_1, \dots, a_N | I, E) &= \prod_{t=1}^N P_{\theta}(a_t | I, E, a_1, \dots, a_{t-1})
 \end{aligned} \tag{5}$$

In this equation,  $a_i$  represents the style attributes of element  $i$ , and  $\theta$  denotes the model parameters. The comparison of the last two rows in Table 2 demonstrates that *NAR* outperforms *AR* across all metrics. This suggests that, unlike machine translation (where *NAR* typically yields lower translation accuracy [31]), the *NAR* pattern is more consistent with poster design.

#### 4.4 Compare AutoPoster with Other Methods

**4.4.1 User Study.** To assess the visual quality of posters, we conducted a survey using a larger-scale questionnaire with settings similar to subsection 4.2. We take the 46 product images as input for test and use AutoPoster to generate posters (referred to as *ours*) without further editing. To ensure a fair comparison, we use the

*Luban*<sup>1</sup> online system without subsequent editing. Additionally, we recruit designers to create generic poster templates. For each product image, we select a random template and replace the background image to obtain a poster, which we refer to as "*Template*". Each product image corresponds to a question, with each question presenting three generated posters. It is worth noting that *Luban* and *Template* require manually inputting suitable taglines, so we use the default ones. To ensure fairness, participants are asked to make choices only based on visual quality (readability and aesthetics). A partial comparison of the produced advertising posters is shown in Figure 8.

In total, 129 participants completed the questionnaire, including 20 professional designers and 109 non-designers. As a result, 5,934 votes were collected. As shown in Fig. 7, the results demonstrate that our method outperforms the other methods in terms of both the  $R1$  (78%) and  $R1 + R2$  (58%) metrics without bells and whistles.

**4.4.2 Online A/B Test.** We conducted an online A/B test in a real e-commerce advertising scenario, randomly dividing online traffic users into two groups. One group viewed ads generated by AutoPoster, while the other group viewed ads created using the *Template* method. We did not use the *Luban* method in this test due to inferior qualitative and quantitative results compared to the *Template* method, which provided a stronger baseline. We tested 80,000 products, collecting around 30 million PV (page views) for each group. The AutoPoster group had a higher CTR (click-through rate) and RPM (Revenue Per Mille) increase of 8.24% and 8.25%, respectively, compared to the *Template* group. These results demonstrate the effectiveness of our method in real industrial applications.

#### 4.5 Poster Generation with Various Sizes

Our system can generate advertising posters of any size from a single input image, thanks to the proposed image cleaning and re-targeting module. Each poster preserves the product's main features, and the graphics position&scale adjusts accordingly, as shown in Figure 9. Unlike cropping-only re-targeting methods, our approach can expand the non-salient region of the original image, creating posters of various sizes. Linearly scaling design elements between posters of different sizes can result in disharmony if the aspect ratio significantly varies, such as from 2:3 to 3:2. However, AutoPoster can perceive these variations in all four design stages, creating posters adaptively for different target sizes.

### 5 CONCLUSION AND FUTURE WORK

In this paper, we introduce AutoPoster, an innovative four-step approach for generating advertising posters that only requires product images, information, and user-specified poster dimensions to create complete posters. Additionally, we propose a novel Style Attribute Predictor that jointly learns multi-attribute prediction and an image re-targeting technique that combines image cleaning to improve diversity and aesthetics in the generated posters. We also provide a large-scale dataset for poster generation. Our experiments demonstrate that AutoPoster and SAP produce posters with superior visual aesthetics compared to existing methods. In future work, we will extend this framework to encompass more forms of information representation design and explore variable font generation.

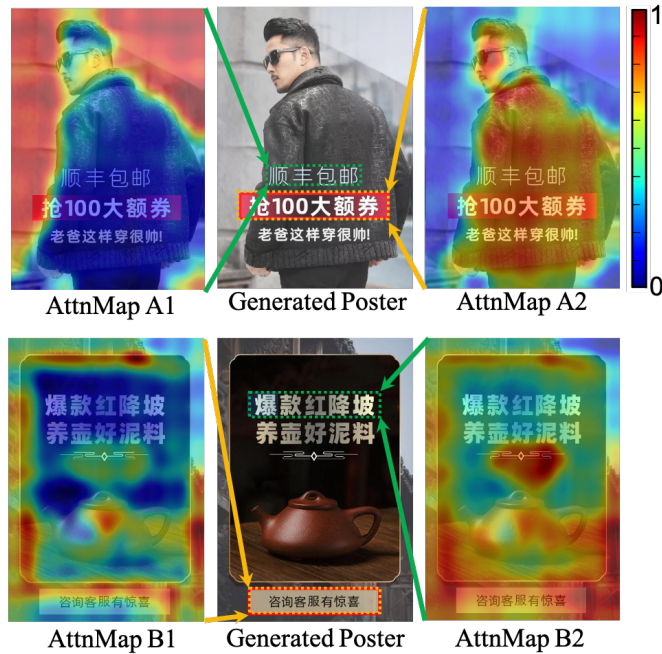
<sup>1</sup><https://luban.aliyun.com>



## REFERENCES

- [1] Shai Avidan and Ariel Shamir. 2007. Seam carving for content-aware image resizing. *ACM Trans. Graph.* 26, 3 (2007), 10. <https://doi.org/10.1145/1276377.1276390>
- [2] 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. *arXiv preprint arXiv:2209.00852* (2022).
- [3] Donghyeon Cho, Jinsun Park, Tae-Hyun Oh, Yu-Wing Tai, and In So Kweon. 2017. Weakly- and Self-Supervised Learning for Content-Aware Deep Image Retargeting. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22–29, 2017*. IEEE Computer Society, 4568–4577. <https://doi.org/10.1109/ICCV.2017.488>
- [4] Saemi Choi, Kiyoharu Aizawa, and Nicu Sebe. 2018. FontMatcher: Font Image Paring for Harmonious Digital Graphic Design. In *23rd International Conference on Intelligent User Interfaces* (Tokyo, Japan) (IUI '18). Association for Computing Machinery, New York, NY, USA, 37–41. <https://doi.org/10.1145/3172944.3173001>
- [5] Xiyang Dai, Yinpeng Chen, Bin Xiao, Dongdong Chen, Mengchen Liu, Lu Yuan, and Lei Zhang. 2021. Dynamic head: Unifying object detection heads with attentions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 7373–7382.
- [6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiuhua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929* (2020).
- [7] Yiqi Gao, Xinglin Hou, Yuanmeng Zhang, Tiezheng Ge, Yuning Jiang, and Peng Wang. 2022. CapOnImage: Context-driven Dense-Captioning on Image. *CoRR* abs/2204.12974 (2022). <https://doi.org/10.48550/arXiv.2204.12974>
- [8] Jiatao Gu, James Bradbury, Caiming Xiong, Victor OK Li, and Richard Socher. 2017. Non-autoregressive neural machine translation. *arXiv preprint arXiv:1711.02281* (2017).
- [9] Shunan Guo, Zhuochen Jin, Fuling Sun, Jingwen Li, Zhaorui Li, Yang Shi, and Nan Cao. 2021. Vinci: An Intelligent Graphic Design System for Generating Advertising Posters. In *CHI '21: CHI Conference on Human Factors in Computing Systems, Virtual Event / Yokohama, Japan, May 8–13, 2021*, Yoshifumi Kitamura, Aaron Quigley, Katherine Isbister, Takeo Igarashi, Pernille Bjørn, and Steven Mark Drucker (Eds.). ACM, 577:1–577:17. <https://doi.org/10.1145/3411764.3445117>
- [10] 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)*.
- [11] Charles E. Jacobs, Wilnot Li, Evan Schrier, David Barger, and David Salesin. 2003. Adaptive grid-based document layout. *ACM Trans. Graph.* 22, 3 (2003), 838–847. <https://doi.org/10.1145/882262.882353>
- [12] Ali Jahanian, Jerry J. Liu, Qian Lin, Daniel R. Tretter, Eamonn O'Brien-Strain, Seungyon Lee, Nicholas P. Lyons, and Jan P. Allebach. 2013. Recommendation system for automatic design of magazine covers. *intelligent user interfaces* (2013).
- [13] Dorothea Jameson and Leo M Hurvich. 1964. Theory of brightness and color contrast in human vision. *Vision research* 4, 1–2 (1964), 135–154.
- [14] Shuhui Jiang, Zhaowen Wang, Aaron Hertzmann, Hailin Jin, and Yun Fu. 2019. Visual font pairing. *IEEE Transactions on Multimedia* 22, 8 (2019), 2086–2097.
- [15] Chuhao Jin, Hongteng Xu, Ruihua Song, and Zhiwu Lu. 2022. Text2Poster: Laying Out Stylized Texts on Retrieved Images. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2022, Virtual and Singapore, 23–27 May 2022*. IEEE, 4823–4827. <https://doi.org/10.1109/ICASSP43922.2022.9747465>
- [16] Dilip Krishnan, Piotr Teterwak, Aaron Sarna, Aaron Maschinot, Ce Liu, David Belanger, and William T. Freeman. 2019. Boundless: Generative Adversarial Networks for Image Extension. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*. IEEE, 10520–10529. <https://doi.org/10.1109/ICCV.2019.01062>
- [17] Ranjitha Kumar, Jerry O. Taltan, Salman Ahmad, and Scott R. Klemmer. 2011. Bricolage: example-based retargeting for web design. In *Proceedings of the International Conference on Human Factors in Computing Systems*. 2197–2206. <https://doi.org/10.1145/1978942.1979262>
- [18] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*. 2980–2988.
- [19] Yuting Qiang, Yanwei Fu, Xiao Yu, Yanwen Guo, Zhi-Hua Zhou, and Leonid Sigal. 2019. Learning to Generate Posters of Scientific Papers by Probabilistic Graphical Models. *J. Comput. Sci. Technol.* 34, 1 (2019), 155–169. <https://doi.org/10.1007/s11390-019-1904-1>
- [20] Evan Schrier, Mira Dontcheva, Charles E. Jacobs, Geraldine Wade, and David Salesin. 2008. Adaptive layout for dynamically aggregated documents. In *Proceedings of the 13th International Conference on Intelligent User Interfaces*. 99–108. <https://doi.org/10.1145/1378773.1378787>
- [21] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. *Advances in neural information processing systems* 27 (2014).
- [22] 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.
- [23] Masataka Tokumaru, Noriaki Muranaka, and Shigeru Imanishi. 2002. Color design support system considering color harmony. In *Proceedings of the 2002 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE'02, Honolulu, Hawaii, USA, May 12 - 17, 2002*. IEEE, 378–383. <https://doi.org/10.1109/FUZZ.2002.1005020>
- [24] Praneetha Vaddamanu, Vinay Aggarwal, Bhanu Prakash Reddy Guda, Balaji Vasanth Srinivasan, and Niyati Chhaya. 2022. Harmonized Banner Creation from Multimodal Design Assets. In *CHI '22: CHI Conference on Human Factors in Computing Systems, New Orleans, LA, USA, 29 April 2022 - 5 May 2022, Extended Abstracts*, Simone D. J. Barbosa, Cliff Lampe, Caroline Appert, and David A. Shamma (Eds.). ACM, 217:1–217:7. <https://doi.org/10.1145/3491101.3519610>
- [25] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).
- [26] Sreekanth Vempati, Korah T Malayil, V Sruthi, and R Sandeep. 2020. Enabling hyper-personalisation: Automated ad creative generation and ranking for fashion e-commerce. In *Fashion Recommender Systems*. Springer, 25–48.
- [27] Sreekanth Vempati, Korah T. Malayil, Sruthi V, and Sandeep R. 2019. Enabling Hyper-Personalisation: Automated Ad Creative Generation and Ranking for Fashion e-Commerce. *CoRR* abs/1908.10139 (2019). [arXiv:1908.10139](http://arxiv.org/abs/1908.10139) <http://arxiv.org/abs/1908.10139>
- [28] Paul A. Viola and Michael J. Jones. 2001. Rapid Object Detection using a Boosted Cascade of Simple Features. In *2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001), with CD-ROM, 8–14 December 2001, Kauai, HI, USA*. IEEE Computer Society, 511–518. <https://doi.org/10.1109/CVPR.2001.990517>
- [29] Bo Wang, Quan Chen, Min Zhou, Zhiqiang Zhang, Xiaogang Jin, and Kun Gai. 2020. Progressive Feature Polishing Network for Salient Object Detection. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7–12, 2020*. AAAI Press, 12128–12135. <https://ojs.aaai.org/index.php/AAAI/article/view/6892>
- [30] Yizhi Wang, Guo Pu, Wenhan Luo, Yexin Wang, Pengfei Xiong, Hongwen Kang, and Zhouhui Lian. 2022. Aesthetic Text Logo Synthesis via Content-Aware Layout Inferring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2436–2445.
- [31] Yisheng Xiao, Lijun Wu, Junliang Guo, Juntao Li, Min Zhang, Tao Qin, and Tiejun Liu. 2022. A Survey on Non-Autoregressive Generation for Neural Machine Translation and Beyond. *CoRR* abs/2204.09269 (2022). <https://doi.org/10.48550/arXiv.2204.09269>
- [32] Xuyong Yang, Tao Mei, Ying-Qing Xu, Yong Rui, and Shipeng Li. 2016. Automatic Generation of Visual-Textual Presentation Layout. *ACM Trans. Multim. Comput. Commun. Appl.* 12, 2 (2016), 33:1–33:22. <https://doi.org/10.1145/2818709>
- [33] Xuyong Yang, Tao Mei, Ying-Qing Xu, Yong Rui, and Shipeng Li. 2016. Automatic Generation of Visual-Textual Presentation Layout. *ACM Transactions on Multimedia Computing Communications and Applications (TOMM)* 12 (March 2016).
- [34] Wenyuan Yin, Tao Mei, and Chang Wen Chen. 2013. Automatic generation of social media snippets for mobile browsing. In *ACM Multimedia Conference, MM '13, Barcelona, Spain, October 21–25, 2013*, Alejandro Jaimes, Nicu Sebe, Nozha Boujemaa, Daniel Gatica-Perez, David A. Shamma, Marcel Worring, and Roger Zimmermann (Eds.). ACM, 927–936. <https://doi.org/10.1145/2502081.2502116>
- [35] Richard Zhang, Phillip Isola, and Alexei A Efros. 2016. Colorful image colorization. In *European conference on computer vision*. Springer, 649–666.
- [36] Nanxuan Zhao, Ying Cao, and Rynson W. H. Lau. 2018. Modeling Fonts in Context: Font Prediction on Web Designs. *Comput. Graph. Forum* 37, 7 (2018), 385–395. <https://doi.org/10.1111/cgf.13576>
- [37] 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 *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23–29 July 2022*, Luc De Raedt (Ed.). ijcai.org, 4995–5001. <https://doi.org/10.24963/ijcai.2022/692>

## A WHAT DOES STYLE ATTRIBUTE PREDICTOR LEARN?

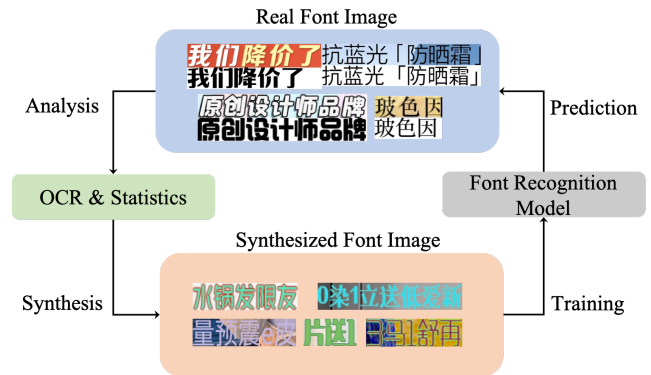


**Figure 10: Visualization of the attention map for SAP decoder. Each poster displays an attention map corresponding to a tagline element (in green line) and an underlay element (in yellow line). The heat map has been normalized to a range of 0 to 1.**

In this section, we attempt to explore the abstract and experiential nature of reasonable color matching between design elements for human designers. In order to gain insights into what the style predictor model learns, we visualize the last cross-attention layer of the style predictor decoder as shown in Fig. 10. AttnMap B2 in Fig. 10 illustrates that SAP has captured the color change of the kettle from black to brown, explaining the prediction of the color gradient from bright yellow to black. An interesting observation was made that despite the lack of any supervised signal about image content, the model implicitly learned to locate and distinguish between the foreground (AttnMap A2 in Fig. 10) and background (AttnMap A1 in Fig. 10) regions of the image. Based on these observations, we conclude that the SAP model has the ability to learn, to some extent, the pattern of color matching based on the appearance of the product subject and the overall color tone style of the image.

## B IDENTIFICATION OF FONT TYPOGRAPHY

To train the style predictor, it is necessary to have labeled attributes of graphic elements as supervisory signals. However, when dealing with taglines on rasterized images, manually identifying the font typography is challenging and remains an open-set problem. To address this issue, we propose a self-supervised training strategy for font typography recognition.



**Figure 11: Font image synthesis and font recognition workflow. For the images displayed in the "Real Font Image", the odd rows are the real font image and the even rows are the font recognized by the model.**

The pipeline can be seen in Fig. 11. Initially, manually designed advertising posters are collected, and Optical Character Recognition (OCR) is employed to recognize the tagline contents in the poster images. Statistical analysis is then conducted on the Chinese character frequency, text size, and tagline length. Subsequently, taglines are rendered on randomly cropped clean background images based on these statistical findings. In the next step, a model is trained using 100k synthesized font images  $I_{syn}$  as the model input, with the ground truth being the font typographies  $f$  used for rendering. The backbone of this model is ResNet-50. It should be noted that the model only recognizes font typographies presented in the synthesized images. For font typographies outside this set, they may be recognized as a similar known one. The final trained model achieves an accuracy rate of 81.8% on a test set containing 5,000 images.

## C POSTER AND ANNOTATION EXAMPLE.

Fig. 12 shows some posters of the constructed dataset mentioned in the paper. Additionally, Fig. 13 provides an example of the poster annotations. The annotation encompasses three key aspects: layout, tagline content, and visual style attribute.

## D LIMITATIONS

Our method has two main limitations. Firstly, the available underlays are currently limited in number and exhibit simplistic styles. This issue may be released by constructing a larger and more diverse library of underlays or introducing supplementary embellishments. Secondly, our method can only predict the color and several font typographies of taglines, lacking the ability to generate variable or artistic fonts at the pixel level (e.g. [30]).



Figure 12: Visualization of a subset of posters extracted from the constructed dataset.

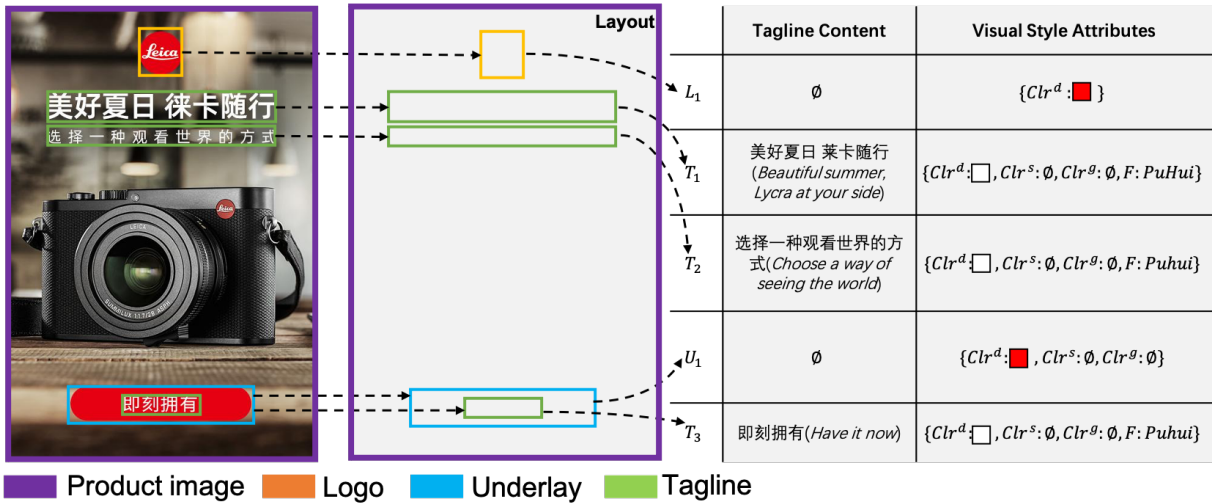


Figure 13: Annotation of an advertising poster. On the left is a poster with Chinese context. The layout of graphic elements on this poster is shown in the middle. And the right part lists the tagline content and visual style attributes of each design element. PuHui is a kind of Chinese font typography.