Harmonious Textual Layout Generation over Natural Images via Deep Aesthetics Learning

Chenhui Li, Peiying Zhang, and Changbo Wang
School of Computer Science and Technology
East China Normal University, Shanghai, China.

***Abstract*—Automatic typography is important because it helps designers avoid highly repetitive tasks and amateur users achieve high-quality textual layout designs. However, there are often many parameters and complicated aesthetic rules that need to be adjusted in automatic typography work. In this paper, we propose an efficient deep aesthetics learning approach to generate harmonious textual layout over natural images, which can be decomposed into two stages, saliency-aware text region proposal and aesthetics-based textual layout selection. Our method incorporates both semantic features and visual perception principles. First, we propose a semantic visual saliency detection network combined with a text region proposal algorithm to generate candidate text anchors with various positions and sizes. Second, a discriminative deep aesthetics scoring model is developed to assess the aesthetic quality of the candidate textual layouts. We build a new Textual Layout Aesthetics dataset with dense annotations of each image and design a reasonable evaluation metric to compare our method with richer baselines. The results demonstrate that our method can generate harmonious textual layouts in various actual scenarios with better performance.**

***Index Terms*—Textual layout, Saliency detection, Image aesthetics, Graphics design, Deep learning**

I. Introduction

IN the field of graphic design, the layout of text and images is common in practical applications such as designing magazine covers, posters, presentations, and packaging design. In particular, with the rise of e-commerce and mobile devices, the demand for graphic layout design is particularly great. Different products usually require different graphic designs. A thousand users require a thousand designs. In the era of artificial intelligence, generating personalized product introductions or advertisements for each mobile phone user has become a new challenge. Traditionally, in the field of visual design, designers frequently invest a high amount of time fulfilling certain repetitive requirements, for example, designing a photograph, adjusting the text layout for different backgrounds and devices, modifying the foreground color, adding the text marks, and evaluating visual effectiveness. In particular, the typography work of text and natural images is very cumbersome. Typography often requires high cost in terms of time, and it is difficult for an inexperienced designer to make higher-level creative designs.

There are many advantages to utilizing an efficient typography design framework. First, it can help designers avoid low-level and highly repetitive tasks while generating high-quality textual layouts. Second, it is conducive to commercial advertising, allowing different users to see different advertising contents. Third, with the increasing number of mobile devices and artificial intelligence technologies, the demand for creative graphic design is increasing. People without art knowledge can use a visual framework to achieve a high-level creative design.

Automatic typography is a practical but challenging problem. Numerous factors, such as image content, text location and size, need to be considered since visual aesthetics rules are complex. In the past, there has been some work on automatic visual design for combining text and image. However, in Text-to-Viz [1], there were many parameters that need to be adjusted for text layout and size selection. Additionally, data-driven methods [2] require amounts of data, and collecting and labeling high-quality data in the visual design area is costly. Previous studies have emphasized in the arrangement of graphic design factors like image elements and text elements. However, on the one hand, text-over-image problem has more constraints. For example, we can hardly change the relation of elements in the background image (e.g., a natural image in bitmap form). On the other hand, exhaustive annotations of each element in an image are costly. The evaluation of the design results is also not mature. It is difficult to judge typography results because textual layout generation is a subjective and flexible task.

Aiming at the above problems, our work focuses on the specific problem of graphic layout: textual layout over a natural image. We propose a deep aesthetics learning approach for placing the text in the right location over the natural image as shown in Figure 1. Inspired by the process of designers first defining an initial textual layout and then adjusting its position and size until selecting the optimal result, our method is composed of two stages, saliency-aware text region proposal and aesthetics-based textual layout selection. Guided by visual perception principles, we take the text position, text size and image content into account. First, we leverage a saliency detection network to model the visual importance of the input image. Then, we adopt diffusion equations to obtain a text-driven probability map and a text-anchor generation algorithm to get candidate text anchors. To capture the perceptual differences between the candidate text regions and select the high aesthetic textual layout results, we extract both the saliency feature and composition feature of different text regions within one image via an efficient deep aesthetics learning model. Several datasets for image aesthetics assessment and image cropping have recently been released [3], [4], but none of them are designed for the text-over-image task. Hence, we construct the Textual Layout Aesthetics (TLA) dataset to train our deep aesthetics learning model and evaluate the performance of our method compared with other baselines. We show multiple practical textual layout design applications enabled by our method. Our contributions are threefold.

(1) We propose a saliency-aware text region proposal method, which can effectively generate a set of candidate text anchors considering the special properties of the text-over-image problem (e.g., text location and size, image content, aesthetics rules).

(2) We build an aesthetics-based deep scoring network to capture the visual perceptual differences between the candidate text regions and select the optimal textual layout result.

(3) We design a set of reasonable evaluation approaches to demonstrate the effectiveness of our method based on the Textual Layout Aesthetics (TLA) dataset. Our method can generate harmonious textual layouts in various actual scenarios.

This work is the extension of our conference version [5]. There are three improvements: (1) Instead of using ResNet [6] to score the candidate textual layouts, we propose a discriminative deep aesthetics model to achieve more accurate layout result. (2) We build a new Textual Layout Aesthetics (TLA) dataset with dense annotations of each image, which facilitates the deep aesthetics learning for textual layouts and method evaluations. (3) We design more reasonable evaluation metrics to compare our method with richer baselines. We provide more practical textual layout applications in various scenarios.

II. Related Work

Our work is related to three aspects of the utilized techniques, namely, automatic graphic design, saliency detection, and image aesthetics assessment.

Automatic Graphic Design

Automatic graphic design layout has been studied extensively. Early works have emphasized in using design templates and aesthetic rules to constrain the layouts. The effectiveness of the optimization approach with aesthetic design rules and visual perception principles has been revealed in [7]. Damera et al. [8] modeled the relations between page elements and addressed the document arrangement problem in probabilistic inferencing over the Bayesian network. Another popular application in graphic design is automated layout design systems, such as a constraint-based recommendation system that can generate the design of magazine covers based on user preferences [9]. Yang et al. [10] designed a system to generate visual-textual presentation layouts with predefined layout templates and aesthetic design principles. With the help of a large database, some learning-based methods are proposed. Gupta et al. [11] used deep learning to generate a dataset of synthetic images of text in a natural way to train a text detection network. Qiang et al. [12] proposed a data-driven framework to generate posters from scientific papers. They used a probabilistic graphical model to predict graphical element attributes and a recursive algorithm for panel layout generation. Micallef et al. [13] utilized perceptual models and quality metrics to enhance the visual quality of scatterplots. Zhao et al. [14] designed a deep learning framework to discover the key design factors influencing upon design personality and score graphic designs with different personalities. Recently, generative adversarial networks (GANs) [15] have shown great success applying to synthesize graphic layout designs. Given a set of initial graphic elements with randomly assigned class probabilities, LayoutGAN [16] proposed a novel wireframe rendering discriminator to distinguish the visual properties of a graphic layout via rendering. The Neural Design Network [17] is a graphic layout generation model that further supported user-specified constraints. Various graphic design layout applications, such as the arrangement of visual and textual contents [2], and indoor scene generation [18] were based on generative models.

In addition to the prior approaches, we propose a deep aesthetics learning framework to generate a harmonious textual layout over a natural image.

Saliency Detection

The saliency detection theory proposed by Itti et al. [19] is the seminal work of the saliency detection approach. These authors believed that the saliency area consists of many points of attention. Later, researchers have explored many efficient methods of saliency detection. Hou et al. [20] redefined the saliency of the image, after which saliency detection is mainly based on regional detection. Subsequent other classical saliency detection methods, such as the AC algorithm [21] and the HC algorithm [22], considered low-level image features like local contrast and color information. Recently, approaches to saliency detection have been moved to deep learning. Relevant work includes a CNN-based visual attention prediction model [23] and an LSTM-based saliency attentive model [24], and a GAN-based saliency detection model [25]. Most of the previous studies have focused on the accuracy of region detection. Qin et al. [26] proposed a hybrid training loss to obtain clear boundaries for salient object detection. Moreover, Bylinskii et al. [27] offered a benchmark framework for saliency detection evaluation.

Although visual saliency is significant in graphic design, aesthetic principles are complex, influenced by many other factors like composition and contrast. Considering only visual saliency may lead to unpleasing layout results. More comparisons and discussions can be found in the work of section IV.

Image Aesthetics Assessment

Early approaches assessed image aesthetics by extracting handcrafted features and classifying the images based on these features [28]. With the development of deep learning methods for image classification, later work achieved significant improvement compared with the traditional approaches. Lu et al. [29] divided images into a global view and local view and used double-column CNNs based on the architecture of AlexNet for classification. Ma et al. [30] proposed a multi-patch aggregation network in which patch selection was based on saliency detection. Inspired by the advancement of the attention model in natural language processing, Sheng et al. [31] proposed attention-based mechanisms to improve the process of multi-patch aggregation. Due to the constraint of fixed-size input of CNNs, the original image is often adjusted by resizing,

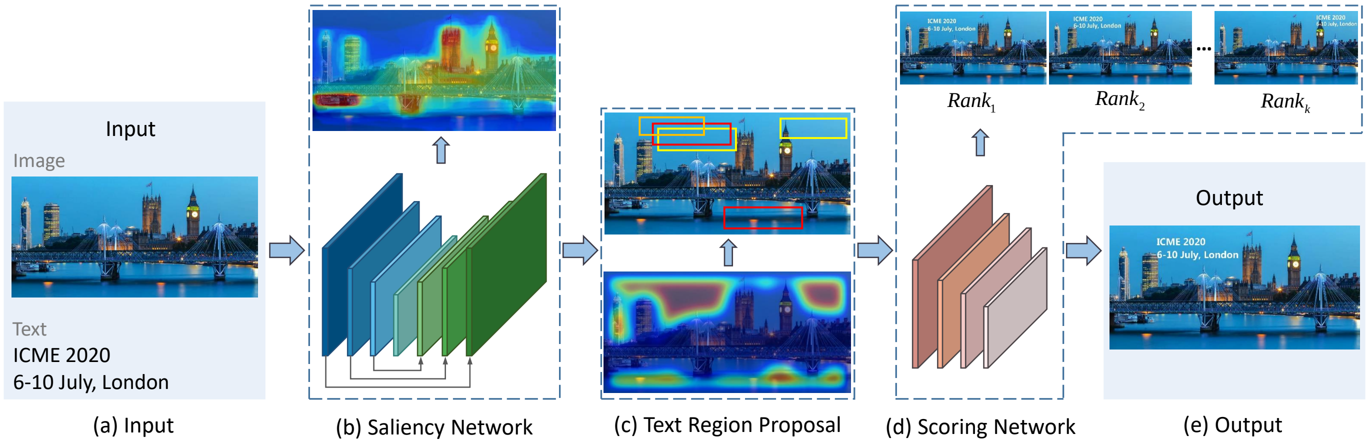


Fig. 1. The structure of our textual layout generation model. We receive a natural image and text as input and output a harmonious textual layout result. Our model has 3 main components. Saliency Network (b): a network to predict the semantic visual saliency map. Text Region Proposal (c): an approach with two algorithms to generate several text anchors with different positions and sizes. Scoring Network (d): a network to assess the aesthetic quality of the candidate textual layouts and output the optimal one.

cropping, or padding, which may destroy the image aesthetics. Mai et al. [32] utilized adaptive spatial pooling layers instead of common layers to avoid image transformations and preserve the image composition quality. Cui et al. [33] utilized distribution over multiple quality levels to predict the aesthetic distributions and aesthetic labels. Inspired by the human visual perception mechanism, Zhang et al. [34] proposed a neural network to capture the holistic information and extract fine-grained features. Image cropping is an important application of image aesthetics assessment. Guo et al. [35] modeled the cropping problem as the least-squares regression problem and proposed a cropping regression method to solve it. Zeng et al. [4] defined more reliable evaluation metrics of the image cropping task. Recently, some studies [36], [37] explored personalized image aesthetics assessment by considering users' social behavior, which reflects their personal perception of aesthetics.

In this paper, we incorporate image aesthetics assessment as a part of our textual layout generation framework. Since textual layout designs have different aesthetic features from common natural images, we design a deep scoring network to find the optimal textual layout result.

III. Methods

A. Overview

Our goal is to generate a harmonious textual layout over a natural image that, given a natural image M_i and text T_i, outputs a harmonious textual layout result MT_o with the optimal text position T_p and size T_s. When designing a typography work of text and natural image, designers first put the text over an appropriate region, and then, adjust the position and size of the initial text layout to obtain the optimal visual effect. With these considerations, our textual layout generation framework can be decomposed into two stages, saliency-aware text region proposal and aesthetics-based textual layout selection. Unlike many other methods in general graphic design, which require original vector data, such as image categorization and element attributes, our models take input images in bitmap form. Our approach has 3 main components as shown in Figure 1:

- *Saliency Network*: a network that receives M_i as input and outputs the semantic visual saliency map S_M.
- *Text Region Proposal*: an approach with two algorithms. One is the diffusion equation, which receives S_M as input and outputs the text-driven probability map PD_M. The other is text anchor generation, which receives PD_M as input and outputs several anchors with different positions and sizes. We can obtain a set of candidate text regions $M_{o_{set}} = \{M_{o_1}, M_{o_2}, ..., M_{o_k}\}$ based on the region proposals above.
- *Scoring Network*: a network that assesses the aesthetic quality of the candidate text regions, which receives $M_{o_{set}}$ as input and outputs the score of each candidate text region. Thus, we can obtain the optimal textual layout result $MT_o \in M_{o_{set}}$.

Before introducing our methods to each section, we define some basic aesthetic rules of the visual layout design.

- *Rule 1* The "text region" should be a rectangular area, which is better not to overlap a complete significant design element, or to cross a strong background image or create discontinuities to offer more harmonious visual effects [11].
- *Rule 2* The "text region" is better in the center of the sub-area and should look symmetric to associated design elements [10].
- *Rule 3* The layout result should gain a higher score in the aesthetic quality assessment.

B. Saliency Map Detection

The saliency map can help us understand the visual importance of different elements in a natural image or graphic design. Inspired by some state-of-the-art saliency detection models, our architecture is based on an encoder-decoder network with a residual refinement module. We refer the reader to BASNet [26] for more details.

The original BASNet is used to detect salient objects, while we aim to obtain a real-valued visual saliency map considering some aesthetic criteria such as *Rule 1*. Human annotations tend to reflect similar relative importance being attributed to a whole design element [38]. Accordingly, to obtain more accurate regional segmentation and clear boundaries, we use a hybrid training loss:

$$L(\Theta) = L_B(\Theta) + L_S(\Theta) \tag{1}$$

where $L_B(\Theta)$ and $L_S(\Theta)$ denote BCEWithLogits loss [39] and SSIM loss [40], Θ is the BASNet model parameters.

Given the ground truth saliency map at each pixel p, $G_{M_p} \in [0,1]$, over all pixels $p = 1, ..., N$, the BCEWithLogits loss is defined as:

$$\begin{aligned} L_B(\Theta) = -\frac{1}{N}\sum_{p=1}^{N}(G_{M_p}\log S_{M_p} + \\ (1-G_{M_p})\log(1-S_{M_p})) \end{aligned} \tag{2}$$

where S_{M_p} is the prediction of the saliency network.

Let x $= \{x_p | p = 1, 2, ..., N\}$ and y $= \{y_p | p = 1, 2, ..., N\}$ be two corresponding image patches extracted from G_{M_p} and S_{M_p}, and let μ_x, μ_y and σ_x^2, σ_y^2 be the mean and variance of x and y, σ_{xy} be their covariance. SSIM loss is defined as:

$$L_S(\Theta) = 1 - \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{3}$$

where $C_1 = 0.01^2$ and $C_2 = 0.03^2$ are scalar constants.

BCEWithLogits loss is used for semantic segmentation on all pixels, and it helps with the accurate regional segmentation. SSIM loss can capture the structural information of the element in an image; hence, it helps clarify boundaries as their weights increase. Our predicted saliency results from the comparison with VisImportance [38] is shown in Figure 8 (b) and (c). The examples demonstrate that our model can predict a more accurate saliency map and clearer boundaries of image elements under various challenging scenarios.

C. Text Region Proposal

After obtaining the original saliency map, to find the optimal text region, a straightforward solution is to list all possible text sizes and positions in the unimportant areas and rank the results. Unfortunately, this solution may cost substantial time on account of the large searching space and it is hard to select the optimal result, since there are many unimportant areas with the same minor saliency values, as areas marked by yellow arrows in Figure 2(a) shows. To handle the problem above, we first develop a diffusion equation model to generate a text-driven probability map, which indicates the probability of text appearing in the corresponding positions. Then, we adopt a text anchor generation algorithm to find the potential text anchors.

Diffusion Equation

The diffusion equation is widely used in image denoising and scale-space image analysis [41]. We utilize the method to generate a text-driven probability map that incorporates the aforementioned *Rule 2*, and the diffusion equation can be described as Equation 4:

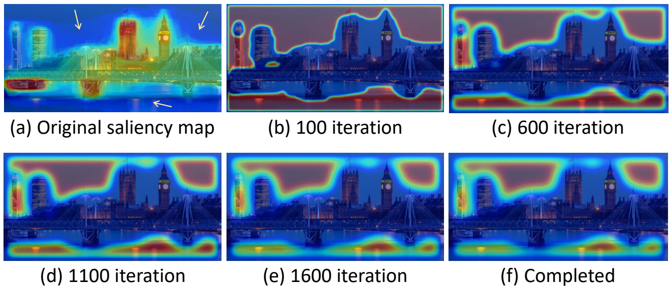


Fig. 2. The iterative process of the diffusion equation. (a) is the original saliency map and (b)-(f) show the process of increasing the number of iterations.

$$\begin{cases} PD_{M+1} = PD_M + \lambda(dX + dY) \\ dX = c_X \nabla_X(PD_M),\ dY = c_Y \nabla_Y(PD_M) \end{cases} \tag{4}$$

where ∇_X, ∇_Y are functions to calculate the gradient of each pixel from horizontal and vertical direction. c_X, c_Y represent the diffusion coefficient of the two directions.

Our diffusion equation aims to generate a text-driven probability map instead of edge-preserving filtering. With this consideration, we therefore set diffusion coefficients c_X, c_Y both to 1 in our implementation. Figure 2 shows an illustrative result. In the initial state of the probability map, there are lots of potential text regions with the same saliency values, which leads to a large set of candidates. The diffusion equation takes into account the visual importance distribution of the graphic elements around each text region. During the iterations, the number of potential text regions is reduced. The iteration stops when the difference between the text-driven probability map and the initial saliency map satisfies a termination threshold. The detailed process of the diffusion equation algorithm is described as Algorithm 1.

Text Anchor Generation

In the object detection area, there are many approaches used to generate bounding boxes, such as selective search [42] and region proposal network (RPN) [43]. However, these methods are designed for object detection, while the proposed anchor boxes are inappropriate for the textual layout generation problem. In image cropping tasks, many methods have been developed for generating candidate crop views [4], [44]. Unfortunately, they focused on the major content or composition patterns of the source image, which are inadequate for our problem. In our specific task, we develop a new text region proposal algorithm based on the text-driven probability map instead of the original image.

Given the text-driven probability map PD_M, we find that candidate regions with high probability values are more likely to be good text regions. Hence, we should generate more candidate anchors in high probability areas. Our algorithm assumes the text region has an aspect ratio $Ratio_T$. First, we divide the important parts of the probability density map into several connected regions and find the peak candidate text anchor of each connected region R_p, with the top-left corner

Algorithm 1 Diffusion equation algorithm.

```
Input: $S_M$: the original saliency map; $\lambda, k$: the scalar constants; $\tau$: the acceptable minimum difference value when iteration completes;
Output: $PD_M$: the text-driven probability density map;
$(W,H) = size(S_M)$, $INF = 255 * k$
$tmp \leftarrow S_M$
for $i = 1$ to $H$ do
    for $j = 1$ to $W$ do
        if $tmp(i,j)$ is important then
            $tmp = INF$
        end if
    end for
end for
while $\Delta val = |PD_M - S_M| > \tau$ do
    for $i = 1$ to $H$ do
        for $j = 1$ to $W$ do
            $\nabla_X \leftarrow$ Calculate gradient of horizontal direction
            $\nabla_Y \leftarrow$ Calculate gradient of vertical direction
            $c_X = 1,\ c_Y = 1$
            $PD_M \leftarrow tmp + \lambda(c_X \nabla_X + c_Y \nabla_Y)$
        end for
    end for
    $tmp \leftarrow PD_M$
end while
$PD_M = 255 - (\frac{PD_M - \min(PD_M)}{\max(PD_M) - \min(PD_M)} * 255)$
return $PD_M$
```

(x_1, y_1) and the bottom-right corner (x_2, y_2). Then, we can generate other candidate anchors based on R_p, and the new top-left corner can be described as:

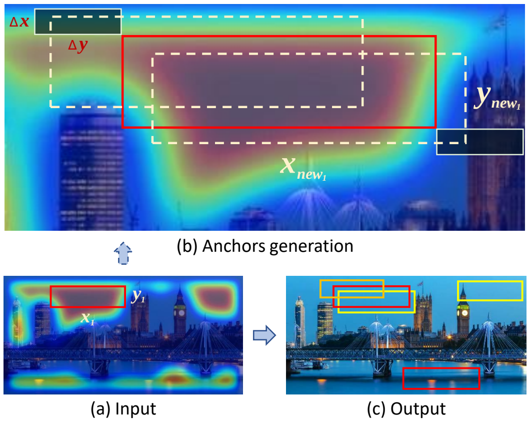
$$\begin{aligned}\{(x_{new_1}, y_{new_1}) | x_1 - \Delta x \leq x_{new_1} \leq x_1 + \Delta x,\\ y_1 - \Delta y \leq y_{new_1} \leq y_1 + \Delta y\}\end{aligned} \tag{5}$$

where $\Delta x = \delta |x_2 - x_1|, \Delta y = Ratio_T \Delta x$ means the adopted range of anchors, and $\delta \propto \sum_{i=1}^{|x_2-x_1|} \sum_{j=1}^{|y_2-y_1|} R_p(i,j)$ defines the deviation coefficient. If R_p is in a higher probability area, the adopted range Δx is larger; that is, there are more candidate anchors generated. R_p can be transformed into multiple scales with the same aspect ratio $Ratio_T$.

After applying shifting and scaling operations on the initial text regions, we obtain a set of candidate text regions. In addition, we can use a grid-based system on images instead of dense pixels to increase search efficiency and reduce redundant candidate text anchors. Figure 3 shows an illustrative result. Algorithm 2 shows the detailed process of the text anchor generation algorithm.

D. Deep Scoring Network

After generating text region proposals, we need to estimate the aesthetic score of each result. A straightforward solution is to train a binary classifier on image aesthetic datasets such as AVA [45] to distinguish between the good and bad textual layout results. However, a general aesthetic classifier is trained across different images [5], and it cannot accurately assess the candidate text regions within one image. To solve this problem, we utilize a data-driven model to capture the perceptual differences between the candidate text regions.



(b) Anchors generation

(a) Input

(c) Output

Fig. 3. Illustration of anchor generation. In (b), the light yellow anchors are generated based on the red anchor R_p, and the black areas show the adopted range of candidate anchors.

Algorithm 2 Text anchor generation algorithm.

```
Input: $PD_M$: the probability density map; $Ratio_T$: the aspect ratio of the input text; $\delta$: the deviation coefficient; $\mu_{\max}, \mu_{\min}$: the maximum and minimum scaling coefficient; $S_{gd}$: the size of a unit grid;
Output: $M_{o_{set}} = \{M_{o_1}, M_{o_2}, ..., M_{o_k}\}$: the candidate text regions;
$W_{gd}, H_{gd} = \frac{size(PD_M)}{S_{gd}}$
$Uni\_set \leftarrow connected\ important\ regions$
$(W_{\min}, H_{\min}) = \frac{(W_{gd}, H_{gd})}{\mu_{\max}}$
$(W_{\max}, H_{\max}) = \frac{(W_{gd}, H_{gd})}{\mu_{\min}}$
for $region$ in $Uni\_set$ do
    for $(w,h) = (W_{\min}, H_{\min})$ to $(W_{\max}, H_{\max})$ do
        $R_p \leftarrow the\ peak\ candidate\ text\ anchor$
        $\Delta x = \delta |x_2 - x_1|$
        $\Delta y = Ratio_T \Delta x$
        $M_{o_{set}}$.append($R_p \pm (\Delta x, \Delta y)$)
    end for
end for
return $M_{o_{set}}$
```

Similar to some popular image cropping methods [4], [46], we build a deep scoring network with a multi-scale feature extraction module and an aesthetic feature extraction module. Figure 4 illustrates the architecture of our deep scoring network. Given an input natural image M_i and a set of candidate text regions $M_{o_{set}}$ (Figure 4(a)), we first apply an expansion operation (Figure 4(b)) on each text region and output the region of expansion (RoE). Each RoE is fed into the multi-scale feature extraction module (Figure 4(c)) to obtain a feature map. With the corresponding text anchor, the aesthetic feature extraction module (Figure 4(d)) uses the RoIAlign [47] and RoEAlign operations to extract the saliency feature and composition feature for each RoE. Finally, the combined feature map is sent through the Fully-Connected (FC) layer to predict an aesthetic score for the candidate text region (Figure 4(e)).

Text Region Expansion

Since the text-level image regions can provide only a few visual features (e.g., similar texture patterns and color themes), we first obtain the expansion of each text region:

$$\begin{aligned} &M_{RoE} = M_i(x_{RoE}, y_{RoE}) \\ &\max(0, x_1 - \alpha H_T) \le x_{RoE} \le \min(H_{M_i}, x_2 + \alpha H_T) \\ &\max(0, y_1 - \alpha W_T) \le y_{RoE} \le \min(W_{M_i}, y_2 + \alpha W_T) \end{aligned} \tag{6}$$

where M_i is the the input image, and H_{M_i} and W_{M_i} are the height and width of the source image M_i. H_T and W_T are the height and width of the text region. α is the expansion coefficient. A higher α corresponds to a larger scale of image contexts. It is worth noting that α is not the higher the better. According to the aforementioned *Rule 2*, there is a tradeoff between the composition features of the expansion sub-area and global features of the source image.

Multi-scale Feature Extraction

Given the region of expansion, we obtain the feature map from a lightweight and efficient backbone network such as ShuffleNetV2 [48], instead of using some classical but complicated pre-trained network such as ResNet101 [6], since aesthetics assessment of the textual layout design does not need to recognize the accurate image categorization or different element attributes [4]. During training, first we resize each input image to 256 × 256. To extract both high-level contexts and low-level details, we upsample and downsample the feature maps from different layers to keep the same size as the input image. Then we use 1×1 convolution to reduce the feature channel dimension and concatenate multi-scale features as the feature map of RoE.

Aesthetic Feature Extraction

At a high level, the goal of the aesthetic feature extraction module is to assess the aesthetic quality of the candidate text regions based on aesthetics rules and select the one with the highest aesthetic score as the final textual layout result. In our textual layout generation task, we need to consider both the saliency feature and the composition feature. On the one hand, the text is better not to overlap a significant design element, or to cross the background region with strong color contrast and complex texture. On the other hand, capturing the composition feature of different textual layouts is important. For example, in Figure 4(b), the fourth text region covers the top of the tower, while the first text region is too close to the edge. Inspired by the recent state-of-the-art image cropping model [4], we adopt the RoIAlign [47] and RoEAlign to extract the aesthetic feature. The deep scoring network can predict aesthetic scores of candidate text regions simultaneously, since it shares convolutional features between different text regions.

First, we extract the text region features from the multi-scale feature map. Then we employ the RoIAlign [47] operation on the text regions with different spatial resolutions of the feature map. The RoIAlign uses bilinear interpolation and average pooling to transform the features of candidate text regions into a fixed-resolution feature vector (denote by F_{RoI}). F_{RoI} can be regarded as the saliency feature of the text region.

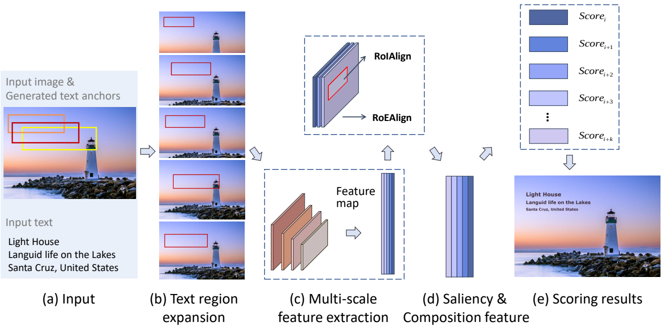

Input image &
Generated text anchors
Input text
Light House
Languid life on the Lakes
Santa Cruz, United States
RoIAlign
RoEAlign
Feature map
$Score_i$
$Score_{i+1}$
$Score_{i+2}$
$Score_{i+3}$
$Score_{i+k}$
Light House
Languid life on the Lakes
Santa Cruz, United States
(a) Input
(b) Text region expansion
(c) Multi-scale feature extraction
(d) Saliency & Composition feature
(e) Scoring results

Fig. 4. The structure of our deep scoring network. The main components are the multi-scale feature extraction module (c) and the aesthetic feature extraction module (d).

Different from image cropping, which can provide rich and discriminative composition features after RoIAlign, the text regions in a graphic design are often with similar uniform color and texture (Figure 4(b)). Another challenge is that the offset of text regions is slight but may have a big impact on the textual layout result. Compared with using the global image features [4], modeling the RoE feature can capture more discriminative composition quality in different candidate textual layouts. Therefore, we introduce the RoEAlign operation to obtain the composition feature. We set the values of the text region to 0 in the feature map of RoE and keep the remaining values unchanged. Similar to RoIAlign, we use the same bilinear interpolation and average pooling to transform the feature map into the same spatial size as F_{RoI} (denote by F_{RoE}). Then F_{RoI} and F_{RoE} are contacted and fed into a FC layer (with a dropout to prevent overfitting), which aggregate saliency features and composition features for final aesthetic score prediction.

Scoring the textual layout result can be modeled as the regression problem. Our deep scoring network is trained using the smooth L_1 loss, which is a widely used loss function in the regression problem with less sensitivity to outliers. Let S_g be the ground truth score and S_p be the predicted aesthetic score of each candidate text region, the smooth L_1 loss is defined as:

$$L_{reg} = \begin{cases} \frac{(S_g - S_p)^2}{2} & \text{if } |S_g - S_p| < 1, \\ |S_g - S_p| - \frac{1}{2} & \text{otherwise.} \end{cases} \tag{7}$$

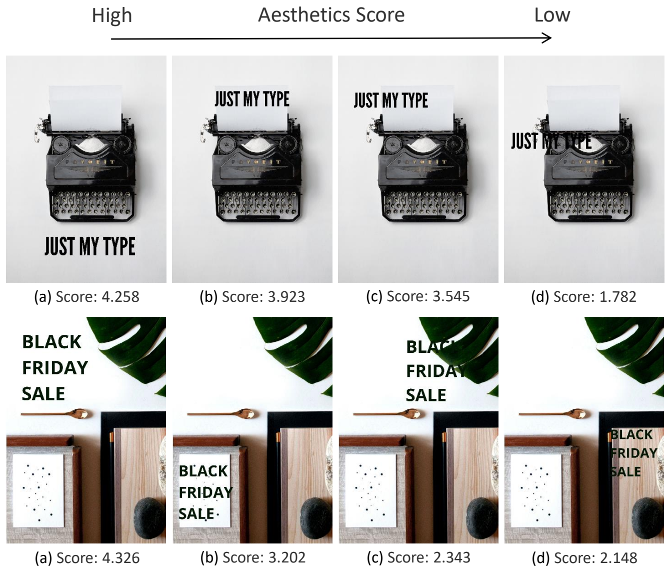
Figure 5 shows some predicted scores using our deep scoring network. The scores are in the range of $[1, 5]$.

IV. Experiment

A. Datasets

Saliency Map Detection

To find the best configuration for the saliency map detection part of our textual layout generation model, we conduct experiments on two different datasets, SALICON (a large-scale dataset collected with the crowdsourcing paradigm) [49] and GDI (visual importance annotations for graphic designs) [38].



(a) Score: 4.258 (b) Score: 3.923 (c) Score: 3.545 (d) Score: 1.782

(a) Score: 4.326 (b) Score: 3.202 (c) Score: 2.343 (d) Score: 2.148

Fig. 5. Ranking textual layout results with different text positions and sizes. Predicted scores are shown below each image.

TABLE I
SALIENCY COMPARISON OF OURS WITH VISIMPORTANCE

Method	Dataset	$CC\uparrow$	$RMSE\downarrow$	$R^2\uparrow$
VisImportance [38]	GDI	0.811	0.181	0.617
Ours	GDI	**0.879**	**0.149**	**0.748**

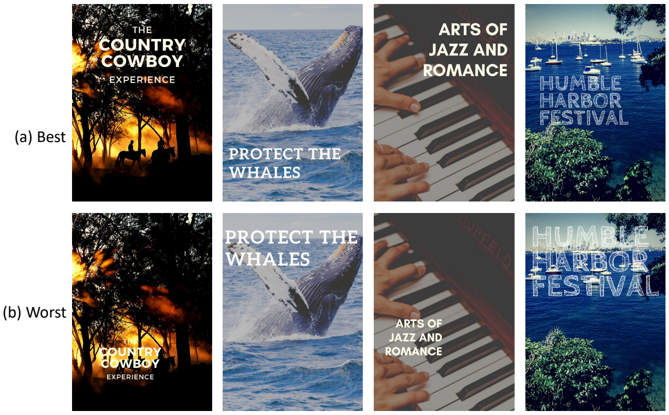


Fig. 6. Our Textual Layout Aesthetics (TLA) dataset consists of various types of design work collected from the design websites (the first row). The second row shows the generated textual layouts with the worst scores.

The experimental results in Table I show a comparison of our saliency network (Ours) with VisImportance [38]. We use the same evaluation metrics as VisImportance for the GDI dataset. Cross Correlation (CC) is commonly used for saliency evaluation. Root-Mean-Square Error ($RMSE$) and the R^2 coefficient measure the correlation between two maps. Higher CC, lower $RMSE$ and higher R^2 are better. Our saliency network improves performance in the GDI dataset compared to VisImportance. The textual layout results are shown in Table II. We observe that GDI outperforms SALICON. The one reason is that the annotations in the GDI dataset are better aligned with the boundaries of image elements. The other reason is that the aesthetic features of textual layout designs are more similar to graphic designs. Therefore, we select the GDI dataset to train our saliency detection model.

Deep Scoring Network

There are many datasets for image aesthetics assessment, such as the AVA dataset [45], which contains score distributions of approximately 255,000 images. We aim to build a discriminative deep aesthetics model to capture the perceptual differences between the candidate textual layouts. However, most of the image aesthetics datasets are curated for the assessment of the composition and aesthetic quality of photographs, which are not applicable to our task of evaluating the textual layout results. Hence, we construct the Textual Layout Aesthetics (TLA) dataset. Compared to some common data collection procedures in image aesthetics assessment, where only a few scores of the overall quality of the source image are annotated, we build our TLA dataset with dense annotations of each image in a way similar to [4].

First, we collect textual layout design works from several poster design websites (e.g., *canva* [50]) in multiple categories (e.g., “Movie Poster” or “Advertising Poster”). Then, we filter the textual layout designs which meet two conditions from the initial data pool: (1) a natural image as the background; (2) a rectangular box as the graphic representation of the textual layout. In poster design websites, graphic elements are editable, so we manually remove other irrelevant graphic elements such as stickers or drawn objects. We separate the background image and text content and save with the metadata (*background image, text content, text position, text size, aesthetic score*). The background images have various aspect ratios and with resolutions of $H \times W$, where $H, W \in [600, 2000]$. Since randomly generating the text regions may achieve most of the obviously poor layout results, we use the aforementioned saliency-aware text region proposal method to generate a set of candidate textual layouts with different scales and positions, which are more likely with good visual effects. The text color and font keep the same as the corresponding initial poster design. We recruit 30 annotators with design or photography experience to give an aesthetic score for the generated textual layouts of each image. Scores range from 1 (worst) to 5 (best). Since the textual layout designs created by professional designers are supposed to well respect visual aesthetic design principles, the scores of the source poster designs from design websites are set to 5, which can be regarded as the best textual layout results in our TLA dataset. In total, we collect 1200 poster designs and each background image has at least 80 candidate text regions with aesthetic scores in our TLA dataset. Figure 6 shows sample images from our textual layout dataset. Taking the Figure 6(b1) as an example, although the color of the cowboy is similar to the dark background, the text should not overlap a significant character element since it may affect viewers’ understanding of the design. We conduct several experiments to show the effectiveness of our TLA dataset by comparing it with some popular image aesthetics datasets.

B. Implementation Details

We use the GDI [38] dataset to train our saliency map detection model, which has 862 training images and 216 testing images. During training, each input image is resized to 256×256 using bilinear interpolation. The initial learning rate is $1e^{-3}$. In deep aesthetics model, we use ShuffleNetV2 [48] for feature extraction, following the setting similar to [4]. We randomly split our TLA dataset into 800 images for training, 200 images for validation and 200 images for testing. Before training, the TLA dataset is applied several data augmentation methods, such as horizontal flipping and adjusting the brightness, contrast and saturation. We used the Adam optimizer [51] with a fixed learning rate $1e^{-4}$ during training. We assign a color to the text based on balancing the contrast score and complementarity score:

$$T_c = \arg\max_{T_c} \alpha Con(T_c) + \beta Com(T_c) \tag{8}$$

where $Con(T_c)$ is the contrast score between the text color T_c and the extracted main color of the corresponding text-level image region, $Com(T_c)$ is the complementarity score between T_c and the colors of its surrounding elements. α is set to 0.7, and β is set to 0.3. The contrast score is calculated according to the Web Content Accessibility Guidelines (WCAG) [52]. For the complementarity score, we utilize the method from O'Donovan et al. [53]. To reduce the search cost, we choose a list of web-friendly colors as candidates instead of the whole color space. We implement experiments on a PC with an Intel Core i7 CPU (with 32GB RAM) and an NVIDIA GeForce 2080 Ti GPU (with 11GB memory). The deep learning framework is based on Pytorch [54].

C. Baselines

Though a number of studies have been developed in image aesthetics assessment and image cropping, few previous methods aim to address the problem of textual layout generation over the natural image. Thus, we design several baselines and compare our deep aesthetics learning framework for textual layout designs with the baselines. All comparisons between different methods are performed on the same testing set of our TLA dataset (200 testing images with the metadata).

Center It is the simplest baseline that put the input text in the center of the background image. We set the size of the text region to $1/8$ time of the input image.

ARKIE It is an existing commercial smart poster design tool [55]. With the input pieces of text and background image, ARKIE can help users generate several candidate poster designs of multiple sizes automatically. It considers some features of the background image such as main color and style, and searches the database to find several similar templates. Then, the text can be placed on the images with proper scales and locations to conform to the templates. To compare our method with ARKIE, we input each testing image and text content to get a recommended layout generation. Since ARKIE uses a set of poster templates, we manually remove graphic elements except for the text element and resize the design result to the original size of the input image.

VisImportance It is a CNN-based model to predict the visual importance maps of graphic designs, which takes the input designs in bitmap form [38]. Though visual saliency is an important factor of image aesthetics, the intrinsic mechanism of aesthetic principles is complicated. For example, other factors such as image composition and image styles play essential roles in the aesthetics assessment of graphic designs. To this end, we train the VisImportance model on the GDI [38] dataset as a baseline to evaluate the performance of considering only visual saliency values in our text-over-image task. First, the VisImportance model takes a testing image as input and predicts a visual importance map. The size of the text region is set to $1/8$ time of the input image. Then, we use a sliding window to find the region with the minimum saliency values and place the text at the corresponding location of the input image.

GAIC It is one of the state-of-the-art models for image cropping [4]. Our pipeline for text-over-image layout generation is proposing candidate text regions at the first stage and then scoring them based on aesthetics assessment, which is similar to the typical methods for image cropping to some extent. However, the aesthetic principles for image cropping are quite distinct from that of textual layout designs, leading to different problem formulations and optimization goals. To compare to GAIC, we directly use their released model trained on the GAICD dataset (an image cropping dataset) to test on our TLA testing set. For further comparison, we use their grid-anchor based strategy to generate candidate regions at the first stage and train their cropping model on our TLA training set to score the candidates.

SmartText It is a prior framework for smart textual layout design [5]. We mainly improve the previous work in the aspect of the deep aesthetics learning model, which boosts the textual layout performance. To evaluate the effectiveness of the improvements, we design a comparison of using SmartText against our extended version. At the first stage, we use the same saliency-aware text region proposal (TRP) method. Then, we follow the steps of SmartText to train a scoring network on our TLA dataset. Specifically, we build a binary classifier based on ResNet101 [6] and use the class probabilities as the aesthetic scores. The TLA dataset is divided into *good* samples (score > 3) and *bad* samples (score ≤ 3) for training. To obtain aesthetic scores, we modify the output layer and reserve the probabilities of *good* class only.

TRP + GIQA It is one of the state-of-the-art methods for image quality assessment [56]. Though GIQA is initially proposed to evaluate the quality of generated images, it can be adopted as a general image aesthetics classifier. To study the effectiveness of our deep aesthetics learning at the textual layout selection stage, we leverage a data-based KNN-GIQA based on *good* textual layout results (score > 3) in the TLA training set. The main idea of KNN-GIQA is to model the probability distribution of *good* samples at first, and then calculate the distance between the testing image and nearby *good* samples in the feature space. We rank the candidate textual layouts based on the quality scores (QS) from KNN-GIQA.

TRP + RoI It is one of the ablation studies to evaluate the performance of different designs for our deep aesthetics learning model. We use only RoIAlign in the aesthetic feature extraction module with the other settings fixed to predict an aesthetic score.

TRP + RoI + RoD In addition, we also evaluate the performance of using the RoIAlign and RoDAlign [4] to extract the aesthetic feature for textual layout designs, where RoD denotes the region of discard. Given an input test image, we first obtain the feature map from the multi-scale feature extraction module. Then we employ the RoIAlign on different text regions in the multi-scale feature map to get RoI feature vectors. RoD features can be constructed by removing RoI features from the feature map and applying the RoDAlign operation. We concatenate the RoI features and RoD features for aesthetic score prediction.

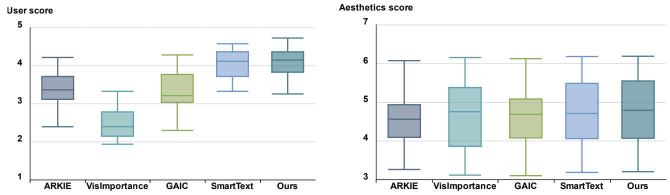


Fig. 7. Results of the visual effect analysis. Left is user scores, and right is aesthetics scores. The results indicate that our method can gain higher scores in both user study and image aesthetics assessment.

D. Evaluation and Discussion

To evaluate the effectiveness of our deep aesthetics learning model for textual layout generation, we design several evaluation methods from three aspects, including benchmark evaluation, image aesthetics assessment, and user study.

Benchmark Evaluation The previous work used the root-mean-square error (RMSE) metric, which measured the differences between two groups of values, to evaluate the performance of textual layout generation models [5]. However, the RMSE metric is unreliable sometimes since it can only indicate the accuracy of text position and size separately. For example, given an input image, the position of the generated textual layout may be similar to the ground truth, while the size differs a lot. On the other hand, RMSE metric is scale-dependent, causing it difficulty to offer a comparison between different datasets. Inspired by evaluation metrics for image cropping and image aesthetics assessment [3], [4], [46], we perform comparisons based on two types of indices.

First, to measure the performance for generating the best textual layout, we use intersection-over-union (IoU) and boundary displacement error (BDE) metrics. IoU is a classical evaluation metric in the object detection area, commonly used for comparing the similarity between two bounding boxes. BDE can measure the displacement between the predicted text regions and ground truth rectangles, defined as follows:

$$BDE = \frac{1}{4}(\frac{|\Delta x_1| + |\Delta x_2|}{H} + \frac{|\Delta y_1| + |\Delta y_2|}{W}) \tag{9}$$

where $\Delta x_k = x_{g_k} - x_{p_k}, \Delta y_k = y_{g_k} - y_{p_k} (k = 1, 2)$. (x_{g_k}, y_{g_k}) denotes the ground truth rectangle and (x_{p_k}, y_{p_k}) denotes the predicted text region. H and W are the height and width of the input image. Higher IoU values and lower BDE values indicate better results.

Graphic design is a quite subjective and flexible task which is difficult to obtain a unique solution. According to Zeng et al. [4], evaluation metrics based on the ranking correlation between the predictions and ground truths are more reasonable. In our TLA testing set, each image has multiple text regions with aesthetic scores. Thus, following Zeng et al. [4], we use the Pearson Correlation Coefficient (PCC), Spearmans Rank-order Correlation Coefficient (SRCC) and K of top-N accuracy ($Acc_{K/N}$) metrics. For each candidate text region, let S_g be the ground truth score and S_p be the predicted aesthetic score, the PCC is defined as:

$$PCC = \frac{\text{cov}(S_g, S_p)}{\sigma_{S_g} \sigma_{S_p}} \tag{10}$$

where cov and σ define the covariance and standard deviation.

The SRCC has a similar definition as the PCC:

$$SRCC = \frac{\text{cov}(R_g, R_p)}{\sigma_{R_g} \sigma_{R_p}} \tag{11}$$

where R_g is the ranking order of ground truth score and R_p is the ranking order of predicted aesthetic score.

$Acc_{K/N}$ can also measure the performance of correctly ranking the candidate text regions, which is defined as:

$$Acc_{K/N} = \frac{1}{K} \sum_{i=1}^{K} True(p_i \in G(N)) \tag{12}$$

where $G(N)$ denotes the ground truth top-N set and p_i is the predicted top-K text region. We calculate $Acc_{4/5}$, $Acc_{4/10}$ and $Acc_{1/1}$ in our experiments.

Image Aesthetics Assessment In natural images, designers tend to put text in well-defined regions, such as regions with uniform color and texture, and good composition. Hence, the text-level image regions should gain high aesthetic scores. With this consideration, we use the NIMA model proposed by Talebi et al. [3] to assess the image aesthetics. NIMA contains a convolutional neural network that can predict the distribution of human opinion aesthetic scores. We crop the text-level image region and feed it into NIMA. Then we can obtain the predicted aesthetic scores for local visual effects.

User Study To evaluate the global visual effects of the textual layout results, we select 50 images from our TLA testing dataset and recruit 30 users to give a score for the 5 methods of generating results: ARKIE [55], VisImportance [38], GAIC [4], SmartText [5] and ours. Scores range from 1 (worst) to 5 (best).

Discussion Figure 7 shows the average aesthetic scores in image aesthetics assessment and user scores in the user study. In the image aesthetics assessment of text-level regions, our model performs favorably against the other methods, since our model tends to place text at the well-defined regions, considering the saliency feature (e.g., background color and texture). The results of the user study show that our method is rated higher than others. As mentioned in subsection III-D, we consider both the saliency feature and composition feature from the deep scoring network. The evaluation results indicate

TABLE II
BENCHMARK EVALUATION

Method	Training Dataset	IoU ↑	BDE ↓	PCC ↑	SRCC ↑	$\text{Acc}_{4/5}$ ↑	$\text{Acc}_{4/10}$ ↑	$\text{Acc}_{1/1}$ ↑
Center	-	0.019	0.239	-	-	-	-	-
ARKIE [55]	-	0.215	0.197	-	-	-	-	-
VisImportance [38]	GDI [38]	0.208	0.203	-	-	-	-	-
GAIC [4]	GAICD [4]	0.073	0.266	0.082	0.113	2.6	5.3	0.08
TRP + ResNet (SmartText [5])	GDI + AVA [45]	0.106	0.259	0.003	0.005	3.4	7.5	0.09
	GDI + TLA	0.295	0.162	0.619	0.635	12.1	22.1	1.79
TRP + GIQA [56]	GDI + AVA	0.097	0.261	0.003	0.004	3.1	7.2	0.09
	GDI + TLA	0.103	0.255	0.230	0.237	4.1	8.7	0.21
TRP + RoI [47]	GDI + TLA	0.476	0.121	0.881	0.856	20.8	36.3	4.09
TRP + RoI + RoD [4]	GDI + TLA	0.496	0.114	0.886	0.864	27.2	43.7	4.68
TRP + RoI + RoE (**Ours**)	SALICON [49] + TLA	0.457	0.118	-	-	-	-	-
	GDI + TLA	**0.529**	**0.109**	**0.888**	**0.867**	**38.4**	**52.6**	**12.28**

that our generated textual layout results can get more harmonious visual effects in both global and local views. Table II shows a comparison of our method and the others in benchmark evaluation. We can find that our deep aesthetics learning model outperforms the others in terms of both generating the best textual layout and rank-order correlation measures.

In Figure 8, we find that ARKIE may simply use templates and lacks consideration of image content. As shown in Table II, ARKIE only achieves comparable performance to the Center baseline. A predefined set of templates are limited and not enough to predict the rich variation of textual layout designs. VisImportance obtains even worse performance than the Center baseline. This is mainly because VisImportance model considers only visual saliency values in our text-over-image task, while other factors such as image composition play indispensable roles in the aesthetics assessment of textual layout designs. If we simply consider visual saliency and put the text on the most unimportant area, the text may be placed too close to the image edge in contrast to some aesthetic rules (Figure 8(d)), leading to unpleasant results. Moreover, VisImportance adopts the original fully convolutional network (FCN) trained on the GDI dataset, while our saliency network helps with more accurate regional saliency results and clearer boundaries of the elements (Figure 8(b-c)). As can be seen, directly using GAIC model trained on the GAICD dataset has unsatisfied performance, while training it on our TLA dataset performs better. It demonstrates that the strategies for image cropping can not be directly applied to our task, since the aesthetic principles and optimization goals for image cropping and textual layout designs are quite different. In addition, our deep aesthetics learning framework for textual layout design is effective, as *Saliency Network* incorporates semantic features and *Text Region Proposal* helps with visual perception principles.

To discover different types of aesthetics features learning, we compare our deep scoring network to SmartText and GIQA. SmartText is based on a general classifier ResNet trained across the good and bad textual layout results. The key idea of GIQA lies in the similarity between different images in a probability distribution perspective, which is designed for image aesthetics assessment. However, both of them cannot accurately evaluate different candidate text regions within one image. We also observe that models trained on the AVA dataset can hardly distinguish the aesthetic quality of candidate textual layouts.

We also conduct ablation studies to further understand the contribution of each module of the deep scoring network. Only using RoI features obtains unsatisfied performance since the features of text regions in a graphic design provide little discriminative information. Using the RoE features achieves better performance than using the RoD features because of the more discriminative composition features. The results in Table II imply the effectiveness of our deep aesthetics learning model.

V. APPLICATIONS

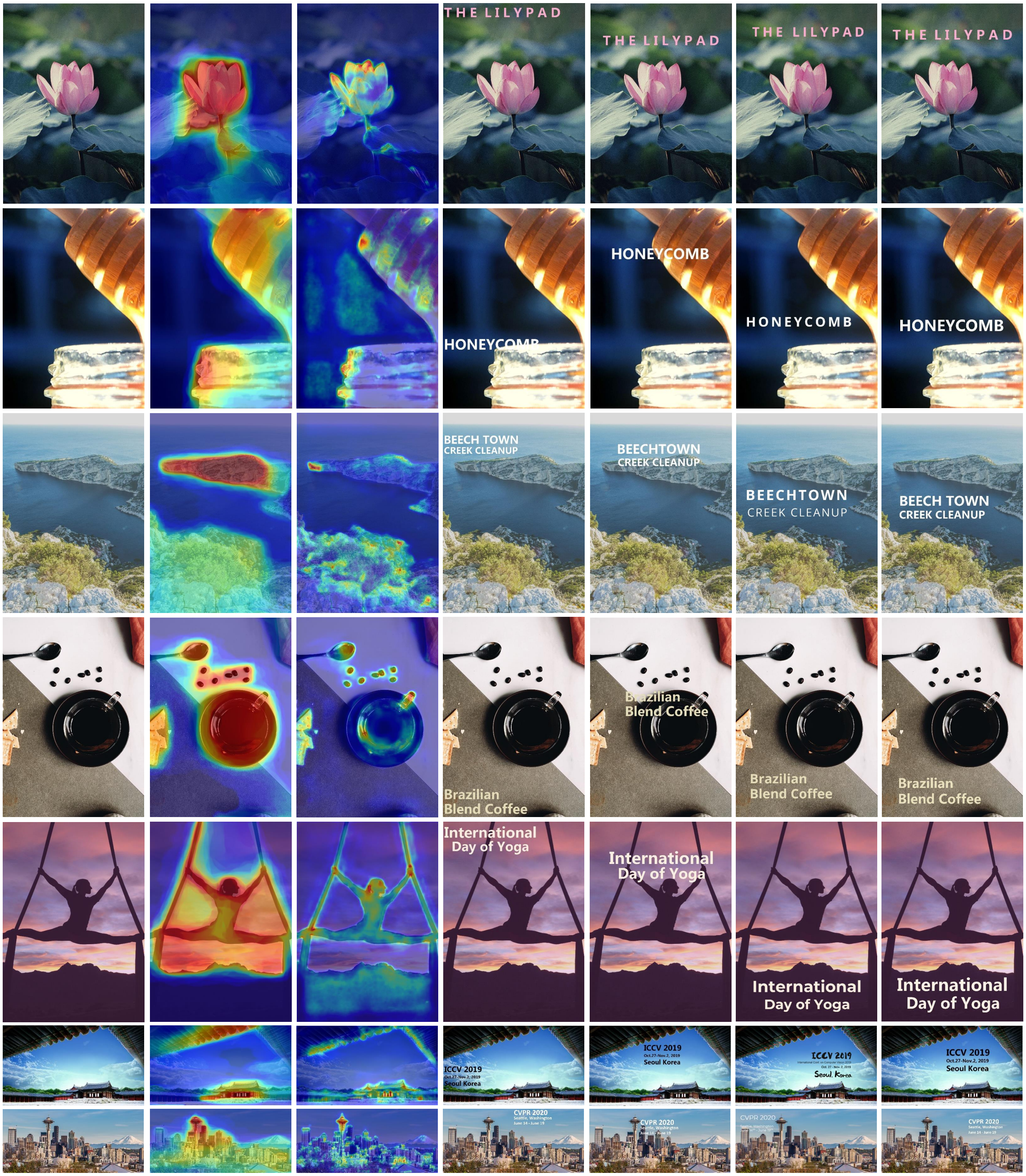
Text with multiple sizes: The acceptable range of lines for users' input text is from one to five lines. Hence, it is necessary to assign multiple sizes to different lines. For example, subtitles are often smaller than the main title in posters. Our method can incorporate the user-specified input as a constraint. If users input n lines of texts, we consider all lines as a whole text anchor, whose ratio is defined as:

$$Ratio_T = \frac{\max(Len(text))}{1 + \sum_{i=2}^{n} \sigma_i} \tag{13}$$

where $\max(Len(text))$ is the maximum length of all text lines, and σ_i is the ratio of the main title to other lines. The results are shown in Figure 8.

Text with mask: In natural images, designers tend to put text in well-defined regions with uniform color and texture. However, in some cases, the background images have strong color contrast and complex texture. Therefore, we propose a method to judge whether it is necessary to apply a mask behind the text, which helps obtain clearer visual effects. We calculate the maximum number of pixels in each connected important region $np_{\max}$ in the text-driven probability map PD_M and use the formula $np_{\max} < \frac{W \times H}{\mu_{\max}}$ to judge, where $\mu_{\max}$ is the acceptable maximum scaling coefficient. $W \times H$ is the resolution of the given natural image M_i. The results are shown in Figure 9.

Text as copyright: Watermark is a common technology to protect the copyright of images. Text embedding in an image



(a) Input (b) Saliency map (**Ours**) (c) Saliency map ([38]) (d) VisImportance (e) ARKIE (f) GT (g) **Ours**

Fig. 8. Textual layout comparison of our method with VisImportance [38], ARKIE [55] and the ground truth. Our method is able to generate harmonious textual layouts in various actual scenarios with better performance, including landscape poster, magazine cover, commodity advertisement and conference poster.

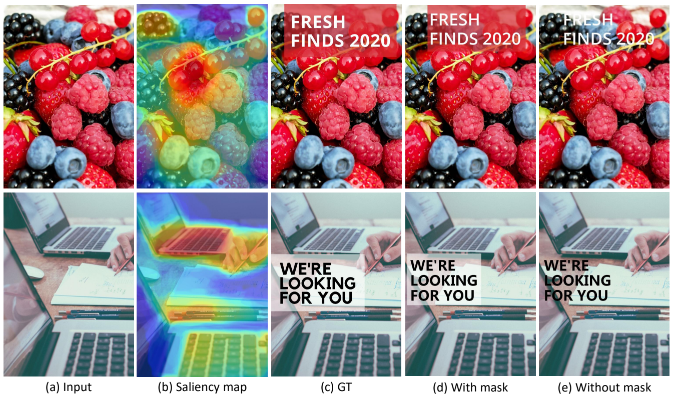


Fig. 9. Text with mask. The mask can help to obtain clearer visual effects when the background image has a strong color contrast and complex texture.

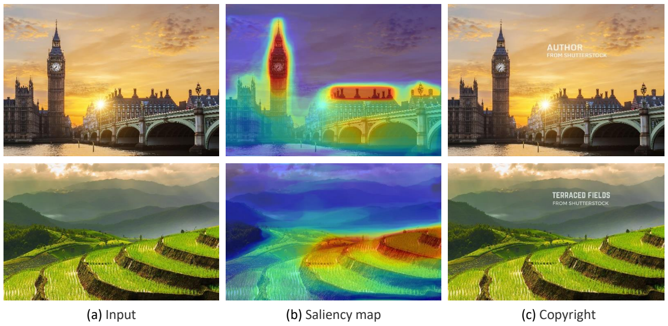


Fig. 10. Text as copyright. It is conducive to copyright protection while keep harmoniously integrated with the image background.

can emphasize the author copyright [57]. Embedded text via an explicit embedding approach is easy to be removed by the image editing technology, and it is also easy to affect the aesthetics of the natural image. Also, information steganography technology [58] has the weak anti-attack ability, and it is easy to lose steganographic copyright information due to partial modification of images. Using our method, it is easy to generate the copyright information harmoniously integrated with the image background. It is conducive to copyright protection while does not affect the overall image quality to a certain extent. It is a new form of copyright as shown in Figure 10.

VI. Conclusions

In this paper, we present a new deep aesthetics learning approach for textual layout generation. We design a feasible framework to place the text in a suitable location according to the requirements of the human visual system. We present a learning-based algorithm to optimize text position placement, combining saliency detection networks with diffusion equations and text region proposals. We develop a deep scoring network to assess the aesthetic quality of the candidate results.

In the future, the presented work can be applied to the poster, magazine cover, and advertisement design. In the video editing work, the placement of subtitles is a challenge. Hence, another potential direction of exploration is to generate text layout for videos. We also plan to further evaluate our method by finding more effective aesthetics assessments and graphic layout approaches.

References

[1] W. Cui, X. Zhang, Y. Wang, H. Huang, B. Chen, L. Fang, H. Zhang, J.-G. Lou, and D. Zhang, "Text-to-viz: Automatic generation of infographics from proportion-related natural language statements," *IEEE TVCG*, vol. 26, no. 1, pp. 906–916, 2019.

[2] X. Zheng, X. Qiao, Y. Cao, and R. W. Lau, "Content-aware generative modeling of graphic design layouts," *ACM TOG*, vol. 38, no. 4, pp. 1–15, 2019.

[3] H. Talebi and P. Milanfar, "Nima: Neural image assessment," *IEEE TIP*, vol. 27, no. 8, pp. 3998–4011, 2018.

[4] H. Zeng, L. Li, Z. Cao, and L. Zhang, "Reliable and efficient image cropping: A grid anchor based approach," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5949–5957.

[5] P. Zhang, C. Li, and C. Wang, "Smarttext: Learning to generate harmonious textual layout over natural image," in *2020 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2020, pp. 1–6.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE CVPR*, 2016.

[7] P. O'Donovan, A. Agarwala, and A. Hertzmann, "Learning layouts for single-pagegraphic designs," *IEEE transactions on visualization and computer graphics*, vol. 20, no. 8, pp. 1200–1213, 2014.

[8] N. Damera-Venkata, J. Bento, and E. O'Brien-Strain, "Probabilistic document model for automated document composition," in *Proceedings of the 11th ACM symposium on Document engineering*, 2011, pp. 3–12.

[9] A. Jahanian, J. Liu, Q. Lin, D. Tretter, E. O'Brien-Strain, S. C. Lee, N. Lyons, and J. Allebach, "Recommendation system for automatic design of magazine covers," in *ACM IUI*, 2013.

[10] X. Yang, T. Mei, Y.-Q. Xu, Y. Rui, and S. Li, "Automatic generation of visual-textual presentation layout," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 12, no. 2, pp. 1–22, 2016.

[11] A. Gupta, A. Vedaldi, and A. Zisserman, "Synthetic data for text localisation in natural images," in *IEEE CVPR*, 2016.

[12] Y. Qiang, Y. Fu, Y. Guo, Z.-H. Zhou, and L. Sigal, "Learning to generate posters of scientific papers," in *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[13] L. Micallef, G. Palmas, A. Oulasvirta, and T. Weinkauf, "Towards perceptual optimization of the visual design of scatterplots," *IEEE transactions on visualization and computer graphics*, vol. 23, no. 6, pp. 1588–1599, 2017.

[14] N. Zhao, Y. Cao, and R. W. Lau, "What characterizes personalities of graphic designs?" *ACM TOG*, vol. 37, no. 4, pp. 1–15, 2018.

[15] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, pp. 2672–2680, 2014.

[16] J. Li, J. Yang, A. Hertzmann, J. Zhang, and T. Xu, "Layoutgan: Generating graphic layouts with wireframe discriminators," *arXiv preprint arXiv:1901.06767*, 2019.

[17] H.-Y. Lee, W. Yang, L. Jiang, M. Le, I. Essa, H. Gong, and M.-H. Yang, "Neural design network: Graphic layout generation with constraints," *ECCV. Springer, Heidelberg*, 2020.

[18] M. Li, A. G. Patil, K. Xu, S. Chaudhuri, O. Khan, A. Shamir, C. Tu, B. Chen, D. Cohen-Or, and H. Zhang, "Grains: Generative recursive autoencoders for indoor scenes," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 2, pp. 1–16, 2019.

[19] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 11, pp. 1254–1259, 1998.

[20] X. Hou and L. Zhang, "Saliency detection: A spectral residual approach," in *IEEE CVPR*, 2007.

[21] R. Achanta, F. Estrada, P. Wils, and S. Süsstrunk, "Salient region detection and segmentation," in *International conference on computer vision systems*. Springer, 2008, pp. 66–75.

[22] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. Torr, and S.-M. Hu, "Global contrast based salient region detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 569–582, 2015.

[23] W. Wang and J. Shen, "Deep visual attention prediction," *IEEE TIP*, vol. 27, no. 5, pp. 2368–2378, 2017.

[24] M. Cornia, L. Baraldi, G. Serra, and R. Cucchiara, “Predicting human eye fixations via an lstm-based saliency attentive model,” *IEEE TIP*, vol. 27, no. 10, pp. 5142–5154, 2018.

[25] J. Pan, C. Canton, K. McGuinness, N. E. O’Connor, J. Torres, E. Sayrol, and X. a. Giro-i Nieto, “Salgan: Visual saliency prediction with generative adversarial networks,” in *arXiv*, January 2017.

[26] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, and M. Jagersand, “Basnet: Boundary-aware salient object detection,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7479–7489.

[27] Z. Bylinskii, T. Judd, A. Oliva, A. Torralba, and F. Durand, “What do different evaluation metrics tell us about saliency models?” *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 3, pp. 740–757, 2019.

[28] Y. Ke, X. Tang, and F. Jing, “The design of high-level features for photo quality assessment,” in *IEEE CVPR*, 2006.

[29] X. Lu, Z. Lin, H. Jin, J. Yang, and J. Z. Wang, “Rapid: Rating pictorial aesthetics using deep learning,” in *Proceedings of the 22nd ACM international conference on Multimedia*, 2014, pp. 457–466.

[30] S. Ma, J. Liu, and C. Wen Chen, “A-lamp: Adaptive layout-aware multi-patch deep convolutional neural network for photo aesthetic assessment,” in *IEEE CVPR*, 2017.

[31] K. Sheng, W. Dong, C. Ma, X. Mei, F. Huang, and B.-G. Hu, “Attention-based multi-patch aggregation for image aesthetic assessment,” in *2018 ACM Multimedia Conference on Multimedia Conference*. ACM, 2018, pp. 879–886.

[32] L. Mai, H. Jin, and F. Liu, “Composition-preserving deep photo aesthetics assessment,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 497–506.

[33] C. Cui, H. Liu, T. Lian, L. Nie, L. Zhu, and Y. Yin, “Distribution-oriented aesthetics assessment with semantic-aware hybrid network,” *IEEE Transactions on Multimedia*, vol. 21, no. 5, pp. 1209–1220, 2018.

[34] X. Zhang, X. Gao, W. Lu, and L. He, “A gated peripheral-foveal convolutional neural network for unified image aesthetic prediction,” *IEEE Transactions on Multimedia*, vol. 21, no. 11, pp. 2815–2826, 2019.

[35] G. Guo, H. Wang, C. Shen, Y. Yan, and H.-Y. M. Liao, “Automatic image cropping for visual aesthetic enhancement using deep neural networks and cascaded regression,” *IEEE Transactions on Multimedia*, vol. 20, no. 8, pp. 2073–2085, 2018.

[36] C. Cui, P. Lin, X. Nie, M. Jian, and Y. Yin, “Social-sensed image aesthetics assessment,” *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 16, no. 3s, pp. 1–19, 2020.

[37] C. Cui, W. Yang, C. Shi, M. Wang, X. Nie, and Y. Yin, “Personalized image quality assessment with social-sensed aesthetic preference,” *Information Sciences*, vol. 512, pp. 780–794, 2020.

[38] Z. Bylinskii, N. W. Kim, P. O’Donovan, S. Alsheikh, S. Madan, H. Pfister, F. Durand, B. Russell, and A. Hertzmann, “Learning visual importance for graphic designs and data visualizations,” in *Proceedings of the 30th Annual ACM symposium on user interface software and technology*, 2017, pp. 57–69.

[39] P.-T. De Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, “A tutorial on the cross-entropy method,” *Annals of operations research*, vol. 134, no. 1, pp. 19–67, 2005.

[40] Z. Wang, E. P. Simoncelli, and A. C. Bovik, “Multiscale structural similarity for image quality assessment,” in *IEEE Asilomar Conference on Signals, Systems & Computers*, 2003.

[41] P. Perona and J. Malik, “Scale-space and edge detection using anisotropic diffusion,” *IEEE TPAMI*, vol. 12, no. 7, pp. 629–639, 1990.

[42] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in *IEEE CVPR*, 2014.

[43] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” in *NeurIPS*, 2015.

[44] Z. Wei, J. Zhang, X. Shen, Z. Lin, R. Mech, M. Hoai, and D. Samaras, “Good view hunting: Learning photo composition from dense view pairs,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5437–5446.

[45] N. Murray, L. Marchesotti, and F. Perronnin, “Ava: A large-scale database for aesthetic visual analysis,” in *IEEE CVPR*, 2012.

[46] Y. Tu, L. Niu, W. Zhao, D. Cheng, and L. Zhang, “Image cropping with composition and saliency aware aesthetic score map.” in *AAAI*, 2020, pp. 12 104–12 111.

[47] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask r-cnn,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[48] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, “Shufflenet v2: Practical guidelines for efficient cnn architecture design,” in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 116–131.

[49] M. Jiang, S. Huang, J. Duan, and Q. Zhao, “Salicon: Saliency in context,” in *IEEE CVPR*, 2015.

[50] Canva, “Canva,” https://www.canva.com, 2020.

[51] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

[52] WCAG, “Wcag 2.0,” https://www.w3.org/TR/2008/REC-WCAG20-20081211, 2008.

[53] P. O’Donovan, A. Agarwala, and A. Hertzmann, “Color compatibility from large datasets,” in *ACM SIGGRAPH 2011 papers*, 2011, pp. 1–12.

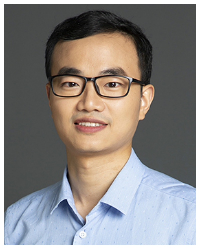
[54] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, “Pytorch: An imperative style, high-performance deep learning library,” in *Advances in neural information processing systems*, 2019, pp. 8026–8037.

[55] ARKIE, “Arkie,” https://www.arkie.cn, 2020.

[56] S. Gu, J. Bao, D. Chen, and F. Wen, “Giqa: Generated image quality assessment,” *arXiv preprint arXiv:2003.08932*, 2020.

[57] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, *Digital watermarking and steganography*. Morgan kaufmann, 2007.

[58] P. Zhang, C. Li, and C. Wang, “Viscode: Embedding information in visualization images using encoder-decoder network,” *IEEE Transactions on Visualization and Computer Graphics*, 2020.



Chenhui Li received Ph.D. from the Department of Computing at Hong Kong Polytechnic University, in 2018. He is an associate professor with the School of Computer Science and Technology at East China Normal University. He received ICCI*CC Best Paper Award (2015) and SIGGRAPH Asia Sym. Vis. Best Paper Award (2017). He has served as a local chair in VINCI2019. He works on the research of computer graphics, information visualization, and multimedia analysis.

Peiying Zhang received her B.Eng. from Nanjing University of Posts and Telecommunications, in 2019. She is working toward a Master degree from East China Normal University, Shanghai, China. Her main research interests include multimedia analysis and information visualization.

Changbo Wang is a professor with the School of Computer Science and Technology, East China Normal University. He received his Ph.D. degree at the State Key Lab of CADCG of Zhejiang University in 2006. He was a post-doctor of the State University of New York in 2010. His research interests mainly include computer graphics, information visualization, visual Analytics, etc. He is serving as the Young AE of Frontiers of Computer Science, and PC member for several international conferences.