ASCII Art Synthesis from Natural Photographs

Xuemiao Xu, Linyuan Zhong, Minshan Xie, Xueting Liu, Jing Qin, and Tien-Tsin Wong

Abstract—While ASCII art is a worldwide popular art form, automatic generating structure-based ASCII art from natural photographs remains challenging. The major challenge lies on extracting the perception-sensitive structure from the natural photographs so that a more concise ASCII art reproduction can be produced based on the structure. However, due to excessive amount of texture in natural photos, extracting perception-sensitive structure is not easy, especially when the structure may be weak and within the texture region. Besides, to fit different target text resolutions, the amount of the extracted structure should also be controllable. To tackle these challenges, we introduce a visual perception mechanism of non-classical receptive field modulation (*non-CRF modulation*) from physiological findings to this ASCII art application, and propose a new model of non-CRF modulation which can better separate the weak structure from the crowded texture, and also better control the scale of texture suppression. Thanks to our non-CRF model, more sensible ASCII art reproduction can be obtained. In addition, to produce more visually appealing ASCII arts, we propose a novel optimization scheme to obtain the optimal placement of proportional-font characters. We apply our method on a rich variety of images, and visually appealing ASCII art can be obtained in all cases.

Index Terms—ASCII art synthesis, non-classical receptive field modulation, texture suppression

1 Introduction

ASCII art is a text-based visual art which presents pictures using printable characters. It gains worldwide popularity in graphic design and communication systems, e.g., advertisement design, icon design, instant messaging systems, and online text-based forums. In particular, structure-based ASCII art gains more attention than tone-based ASCII art due to its concise and compact text arrangement and adaptivity to small-screen devices. But manually creating structure-based ASCII art is not easy; the artists need to meticulously select and position every single character, which is quite tedious and time-consuming. To facilitate the automatic generation of structure-based ASCII art, methods have been proposed to convert line drawings to ASCII art [1]. However, to create ASCII art from natural photographs, one need to first identify the main structure from the photographs, and then use text characters to represent the identified main structure. But extracting sensible structure from natural photographs without excessive amount of undesirable details is extremely challenging.

The main challenge is to extract the structure lines that are meaningful to human perception. Here, we observe that when creating ASCII art from a natural photograph, the artists can intelligently drop the detailed textures and preserve only the main structure in the created art (Fig. 1b). Therefore, standard edge detection methods are generally not applicable since they tend to extract all lines from the input image, without caring the lines are from the main structure or texture. In this case, these excessively extracted texture lines lead to unpleasant or even unrecognizable results (e.g., Fig. 2b). Our second observation is that, according to different target ASCII art resolutions, the artists adapt their works by dropping different amounts of texture details. For example, the artist drops almost all the textures in Fig. 1b due to the small resolution, but keeps the large-scale textures in Fig. 1c with a larger resolution. Therefore, the extracted structure should be controllable and adapted to the desired target resolution. Our last observation is that, the main structure inside the photographs should always be preserved no matter how weak the structure is (e.g., the antenna of the butterfly in Fig. 1 and the legs of the crane in Fig. 2). However, while humans can easily identify these weak structure lines, the existing texture filters [2], [3], [4] and contour detectors [5], [6] are unable to preserve them as they are removed together with the surrounding texture which has a relatively stronger contrast (Figs. 2c, 2d, and 2e). Effective preservation of the main structure with weak constrast from the texture region is a key challenge in generating sensible ASCII art from natural photographs.

In summary, we need a texture-aware line extraction method to remove the unwanted texture so that the generated ASCII art is neat and concise. In addition, to suit for different resolution requirement, the amount of the extracted structure lines (or the removed texture lines) should be controllable. Finally, our structure line extraction method should be sensitive to human perception so that weak-constrast structure within the strong-constrast texture can be identified as well. To achieve these goals, we propose to model the human perception of structure-from-texture using the *non-classical receptive field (non-CRF)* modulation [7]. The non-CRF is a modulation of the texture suppression mechanism of human visual system

- *X. Xu is with the School of Computer Science and Engineering, South China University of Technology, Guangzhou, China. E-mail: xuemx@scut.edu.cn.*
- *L. Zhong and M. Xie are with South China University of Technology, GuangZhou, China. E-mail: {z.linyuan, csmsxie}@mail.scut.edu.cn.*
- *X. Liu and T.-T. Wong are with the Chinese University of Hong Kong, and Shenzhen Research Institute of the Chinese University of Hong Kong, Hong Kong. E-mail: {xtliu, ttwong}@cse.cuhk.edu.hk.*
- *J. Qin is with School of Nursing, the Hong Kong Polytechnic University, Hong Kong. E-mail: jqin@szu.edu.cn.*

Manuscript received 3 Sept. 2015; revised 1 May 2016; accepted 3 May 2016. Date of publication 14 June 2016; date of current version 30 June 2017. Recommended for acceptance by K. Myszkowski. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below. Digital Object Identifier no. 10.1109/TVCG.2016.2569084

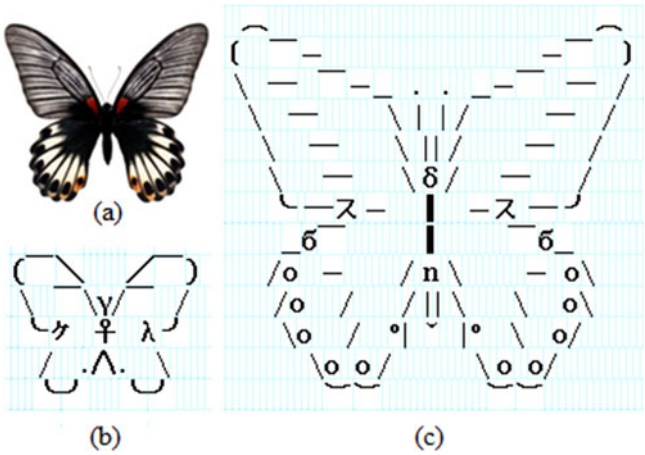


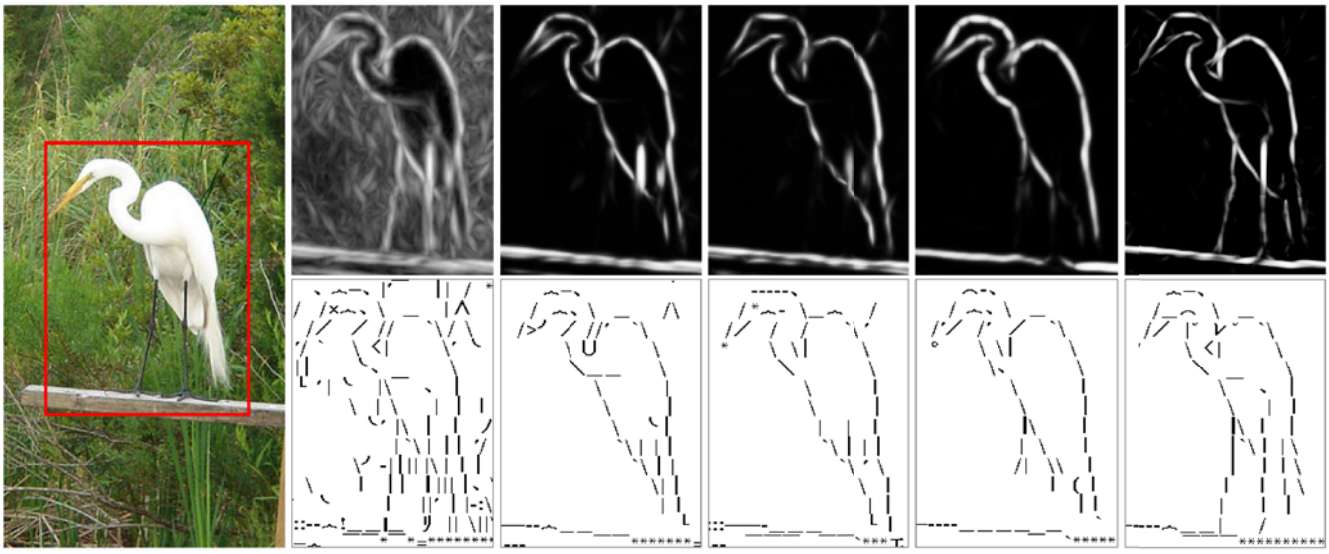
Fig. 1. (a) Input image. (b) & (c) are ASCII arts created by an artist based on two different target text resolutions (5 and 13 rows respectively).

in physiology, and it has two key properties that benefits our system. One is its scale-aware texture suppression ability which allows us to control the amount of the removed texture based on different target resolution. The other is its orientation selectivity which can help us distinguish weak-contrast structure from texture. In fact, the mechanism of non-CRF modulation can be modeled in various ways [8], [9], [10], [11], but the existing models generally tend to remove most of the textures for contour detection. As a result, weak-contrast structure within the texture may also be undesirably removed, and the controllability over the amount of texture to remove is also insufficient. Hence, we propose a new model of non-CRF modulation which can directly control the *scale of texture suppression* (the amount of texture to suppress). Besides, our model can better distinguish weak-contrast structure within the texture region.

To do so, we first compute a structure map from the input photograph via a multi-scale CRF response simulation model. We employ a newly proposed multi-orientation phase congruency model [12] to simulate the multi-scale CRF response. With this model, we could extract the CRF response of different scales in aware of orientation. Based on the CRF response, we further propose a novel non-CRF modulation scheme that incorporates a *texture asymmetry* term to preserve the structure that forms the boundary of texture region. Next, we further match the extracted structure lines with text characters. Similar with [12], we also adopt proportional-font characters so that the created ASCII arts look more concise. But different from Xu's method which only optimizes the character arrangement in each row separately, we propose an optimization scheme which optimizes the character arrangement not only inside each row, but also between every two neighboring rows. This approach ensures the smoothness and the consistency of generated ASCII art in both horizontal and vertical directions.

To demonstrate the effectiveness of our method in structure extraction and texture suppression, we compare our structure extraction method to three state-of-the-art texture filtering methods [2], [3], [4] and the gPb contour detector [5]. The experiments show that our method not only achieves better control over the scale of texture suppression, but also more consistent to human perception. We further compare our ASCII art generation method to the existing ASCII art generators [1], [12] where our results look much more visually pleasant than those generated by other methods. Our contributions can be summarized as followed.

- We introduce the *non-CRF modulation* mechanism to the application of ASCII art generation. Furthermore, we propose a new model of non-CRF modulation tailored for ASCII art generation. Our model of non-CRF modulation not only better controls the scale of texture suppression, but also better preserves perception-sensitive weak structure within the texture region.
- We propose a novel optimization scheme to match the structure lines with an optimal arrangement of proportional-font characters.



(a) Input (b) Xu et al.[2] (c) Zhang et al.[3] (d) Cho et al.[4] (e) gPb [5] (f) Our method

Fig. 2. Our structure extraction method successfully identifies the legs of the crane as structures while removing the excessive grass textures.

2 Related Work

Here, we first review existing ASCII art generation methods. As two main steps to synthesize ASCII art from natural photographs are contour detection and texture smoothing, we also review key works that relates to our application.

2.1 ASCII Art Synthesis

Most existing ASCII art generation methods accept only line drawing as input [1], [13], [14], [15], to avoid the complication of contour detection. To generate ASCII arts from natural photographs, Xu et al. [12] proposed to first extract the features of the input image using a multi-orientation phase congruency model, and then match these features during ASCII art generation. However, since their method extract all sensible structures from input, the generated ASCII art can be overcrowded with excessive structure, especially when the target resolution is small. The reason behind such excessive structure is the lack of texture suppression. Furthermore, their character placement pays no attention to neighboring rows, and may cause visual artifact when there exists contour near the text row boundary. In contrast, our method can control the scale of the texture being suppressed, and we pay more attention on the boundary between text rows to minimize the potential visual artifact. Recently, a novel technique based on classification has been proposed to achieve fast rendering of image mosaics and ASCII Art [16]. However, this method focuses mainly on speeding up the ASCII Art generation and image mosaic rendering, and the conciseness of their results is doubtful. In contrast, our work focuses more on conciseness of the ASCII arts.

2.2 Contour Detection

Classical edge detectors (e.g., Canny and Gabor) extract edges solely based on local contrast, so they are unable to distinguish structural lines and texture lines. Note that whether a line is a texture line sometimes is scale-dependent. When the texture is observed in a large scale, we may perceive the texture line as structural line. In other words, to distinguish structural lines and texture lines, the notion of scale is needed. Some state-of-the-art contour detectors could compute structure maps of different levels of details. Kang et al. [17] proposed a 1D flow-based DoG filter to extract the structural lines under different scales, but their method cannot be directly extended to 2D texture. Arbelaez et al. [5] proposed to extract key structure and suppress fine texture by combining local cues in multiple channels into a global optimization framework based on spectral clustering. Dollar et al. [6] exploited the advanced machine learning techniques for structure extraction. Though their method could generate good results, the success of machine-learning based approaches heavily depends on the preparation of extensive training dataset.

2.3 Texture Smoothing

Recently, several structure-aware texture smoothing methods have been proposed to suppress texture while preserving structure. Xu et al. [2] proposed the relative total variation (RTV) metric to smooth textures, and their method is based on an assumption that texture pixels should have much higher RTV values than structure line/edge pixels. Karacan et al. [18] proposed an image smoothing approach depending on covariance matrices of simple image features to implicitly capture local structure and texture information. Zhang et al. [3] further proposed to smooth texture with the control of details under a certain scale, and they intended to completely remove all small-scale texture while preserving large-scale structure (though maybe blurred). Recently, Cho et al. [4] proposed a bilateral texture filter to suppress texture with fine scale and relative low contrast while preserving key structure. Aubry et al. [19] proposed a variant of the bilateral filter that produces cleaner edges while retaining its speed. However, all the above methods generally suffer from the inability to distinguish weak-contrast structure from strong-contrast texture due to their filtering nature. Since these methods are all filtering methods, they are essentially sensitive to local contrast. Therefore, the weak-contrast structure are very likely to be mistakenly removed in order to suppress strong-contrast texture. Besides, since there is no consideration of the line orientation, the structural lines within a texture are also likely to be removed if their local contrasts are similar. On the other hand, our method incorporate a perception- motivated model that allows us to extract structure lines even if the contrast of structure lines is weaker than or similar to that of the surrounding texture.

3 Overview

Our system flow is shown in Fig. 3. It mainly includes two phases: structure extraction and text-placement optimization. The input of our system is a natural photograph and a character library including the user- selected fonts. Note that both fixed-width fonts and variable-width fonts (proportional fonts) are supported in our system. Our system first computes the structure maps for the input photograph and each character (Fig. 3b). Note that the structure map computation of each character is a preprocessing step and is performed only once. We propose a new structure extraction scheme that can extract weak-contrast structure while suppressing texture (detailed in Section 4). Each computed structure map contains both magnitude and orientation information.

With the computed structure maps, we can then match the structural lines of input with that of text characters (Fig. 3c), in order to determine the character for reproducing the original image content. Given the user-specified text resolution, the structure map of the input is scaled to match with that of the characters. Here, the user only needs to determine the height (number of rows), and the width (number of characters inside each row) is automatic determined by preserving the aspect ratio of the input image. During the matching, we perform a novel optimization which optimizes the character placement along the horizontal direction and minimizes the chance of potential artifact between neighboring rows (Section 5).

4 Structure Extraction

In order to simulate human perception, our structure extraction scheme should be structure-sensitive and texture-suppression in a scale-controllable manner. Based on the physiological studies of visual cortex [7], [20], the *non-CRF*

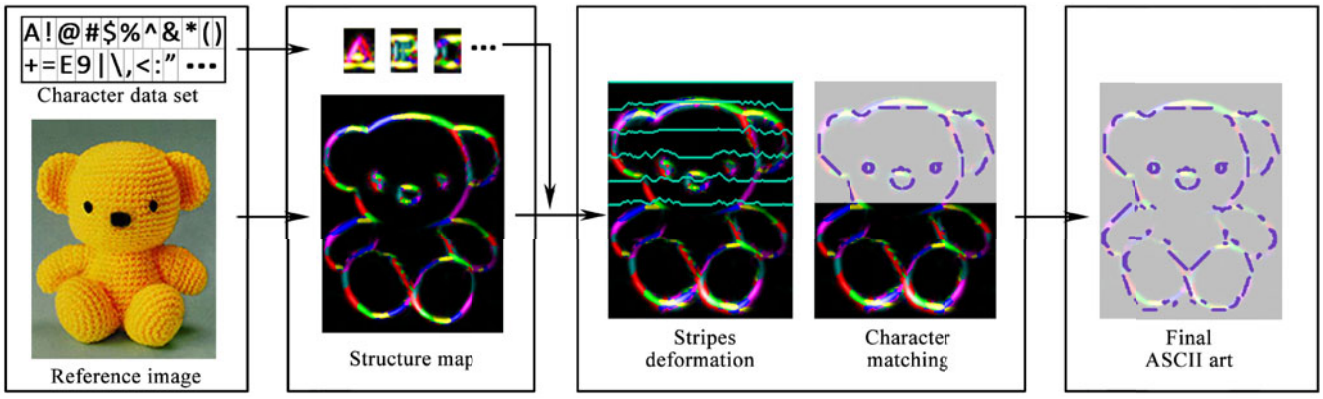


Fig. 3. System overview.

modulation mechanism of human visual perception plays an important role in distinguishing salient structure from fine details, such as noise and small-scale textures. This non-CRF modulation mechanism in fact is quite suitable for our application, due to its structure-sensitivity and scale-controllability. In fact, the non-CRF modulation mechanism has already been studied and modeled in different previous applications [8], [9], [11], [21], [22], [23]. However, all existing models tend to remove texture indifferently and lack of control over the scale of texture being suppressed. In this paper, we propose a new model of non-CRF modulation which can better control the scale of texture suppression and also better distinguish weak-contrast structure from the surrounding texture.

In the rest of this section, we first give a brief introduction of the non-CRF modulation (Section 4.1). Then, we introduce the traditional non-CRF models and analyze their shortcomings (Section 4.2). We present our model of non-CRF modulation in Section 4.3.

4.1 Receptive Field Structure

The modeling of non-CRF modulation is usually based on a specific receptive field (RF) structure. Among several psychological RF structures, two of them receive much attention from the computer vision community: the annular structure [22], [24] and the butterfly-shaped structure [11], [21], [23]. In our model, we also employ the butterfly-shaped structure as it can avoid self-inhibition [11]. The butterfly-shaped RF structure is shown in Fig. 4 (the top-right blue dotted box). The receptive disk can be divided into an inner disk (classical receptive field, CRF) and a butterfly-shaped outer ring (non-classical receptive field, non-CRF). From psychology theory, the stimuli in the non-CRF regions (green regions) inhibit the neural response to similar stimuli in the CRF region (white region). Moreover, if the stimuli in the non-CRF regions are closer to the center of the disk, the inhibition of the neural response to the stimuli in the CRF region should be stronger, which is illustrated by the attenuation of luminance in the green region. Here, β controls the opening angle of the wings and it is commonly set to 60 degree. The modeling of non-CRF modulation involves two key design issues: simulating the CRF response and the non-CRF modulation.

4.2 Traditional Non-CRF Modulation

Traditional models of non-CRF modulation simulate the CRF response using a single-scale Gabor filter, which has been proved to be consistent with the human perception under a specific scale. For an input image $I(p)$, the Gabor filtering of a pixel p under scale (standard deviation) σ at orientation θ can be presented by:

$$\begin{aligned}
&[g^r_{\sigma,\theta}(p), g^i_{\sigma,\theta}(p)] = [I(p) * f^r_{\sigma,\theta}(p), I(p) * f^i_{\sigma,\theta}(p)]\\
&f^r_{\sigma,\theta}(p) = e^{-\frac{x'^2+\gamma^2 y'^2}{2\sigma^2}} \cos\left(2\pi\frac{x'}{\lambda}+\varphi\right)\\
&f^i_{\sigma,\theta}(p) = e^{-\frac{x'^2+\gamma^2 y'^2}{2\sigma^2}} \sin\left(2\pi\frac{x'}{\lambda}+\varphi\right)\\
&x' = x\cos\theta + y\sin\theta\\
&y' = -x\sin\theta + y\cos\theta,
\end{aligned} \tag{1}$$

where $f^r_{\sigma,\theta}(p)$ and $f^i_{\sigma,\theta}(p)$ correspond to the real and imaginary parts of Gabor filter; (x, y) is the 2D coordinate of pixel p; $\gamma = 0.5$ is a constant called spatial aspect ratio; σ/λ determines the spatial frequency bandwidth, and it is set as 0.56 in our work; φ is a phase offset and defaulted by 0 in our work; $*$ is the convolution operator.

Generally, its local energy is adopted to reflect the image structure, and defined as:

$$G_{\sigma,\theta}(p) = \sqrt{(g^r_{\sigma,\theta}(p))^2 + (g^i_{\sigma,\theta}(p))^2}. \tag{2}$$

Then, the map of CRF response can be inhibited by

$$G'_{\xi,\theta}(p) = G_{\sigma,\theta}(p) - \alpha \cdot \Psi(p;\xi,k,\theta) * G_{\sigma,\theta}(p), \tag{3}$$

where α is a weighting factor to control the strength of the inhibition effect; $*$ is the convolution operator; Ψ is a weight template for the RF structure and the CRF response map G; ξ represents the scale of the non-CRF region (usually using the inner radius of the butterfly-shaped ring); k is the ratio between the inner radius and the outer radius of the butterfly-shaped ring. Note that the preferred orientation of the RF structure is actually consistent with the orientation of Gabor filtering θ. Hence, Ψ can be defined as:

$$\Psi(p;\xi,k,\theta) = |\Psi_l(p;\xi,k,\theta) + \Psi_r(p;\xi,k,\theta)|. \tag{4}$$

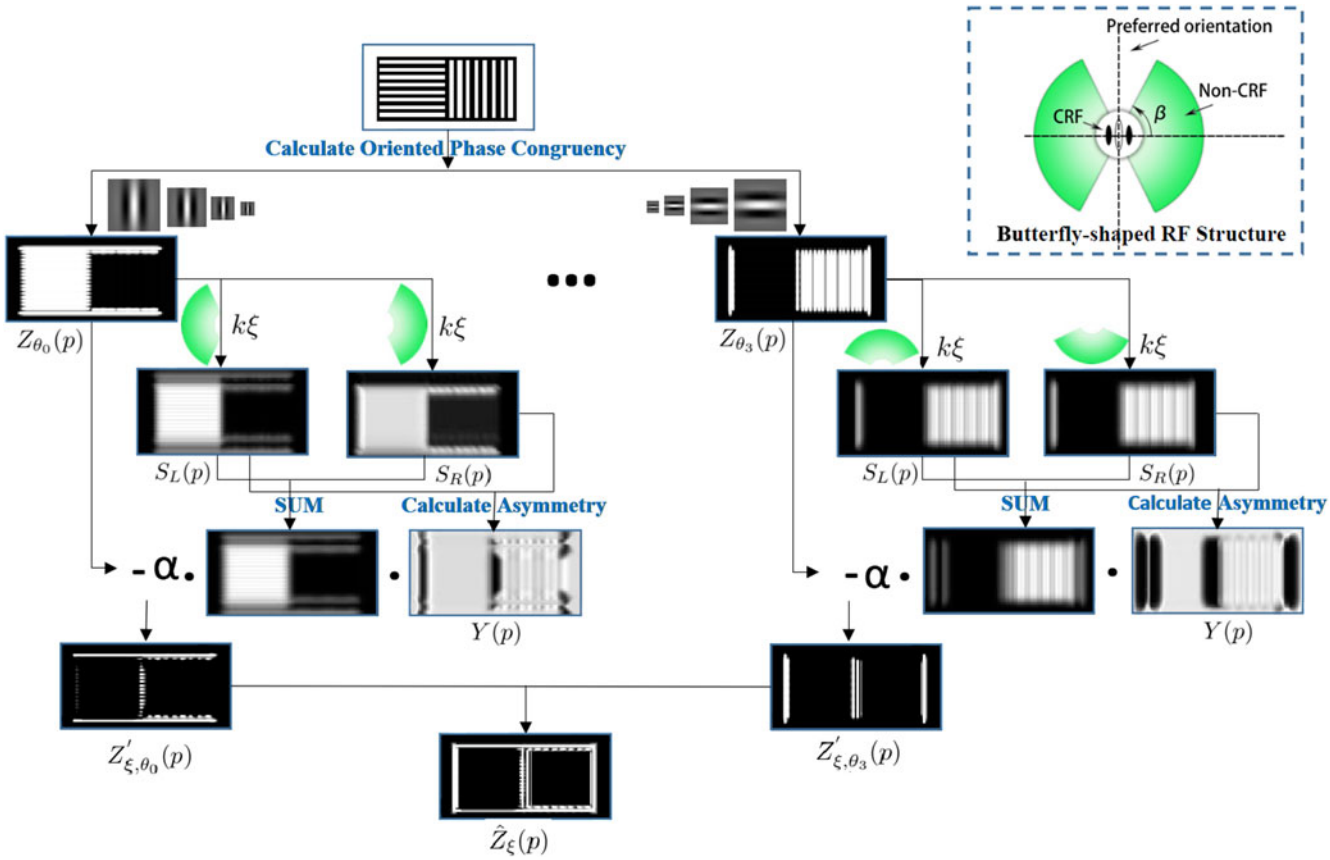


Fig. 4. The flow of our modelling of non-CRF modulation.

Here, $|\cdot|$ denotes the absolute value; Ψ_l and Ψ_r are the weighting functions of the left and right wings of the non-CRF region in the butterfly-shaped structure:

$$\Psi_l(p;\xi,k,\theta) = \begin{cases} \max\{DOG(p;\xi,k),0\} & -\beta \le \omega+\theta \le \beta \\ 0 & o.w. \end{cases}$$
$$\Psi_r(p;\xi,k,\theta) = \begin{cases} \max\{DOG(p;\xi,k),0\} & -\beta \le \omega+\theta-\pi \le \beta \\ 0 & o.w. \end{cases} \quad (5)$$

Here, the $DOG(p;\xi,k)$ is a centered difference of the Gaussian functions with the two scales of ξ and $k\xi$. We employ the positive values of the DOG to approximate the non-CRF modulation mechanism, as the pixels closer to the CRF region have larger DOG values and hence produce stronger inhibition effect. ω is the angle of p in the polar coordinate system. We set $k=4$ in our experiments, as the radius of non-CRF is normally about 2 to 5 times larger than that of CRF [11]. Note that in the traditional model, ξ is usually set equal to σ, which is the scale of the Gabor filter.

Lastly, the modulated result can be obtained by selecting the maximum magnitude after inhibition among all orientations, $\hat{G}_\xi(p) = \max_{j\in[0,\kappa-1]} G'_{\xi,\theta_j}(p)$, where κ is the total number of orientations.

In this traditional model of non-CRF modulation, the scale for texture suppression can somehow be adjusted by tuning the parameter ξ since it determines the range of the non-CRF region. Moreover, it binds the scale for texture suppression and the scale for simulating CRF response by setting $\xi=\sigma$. Fig. 5b,c,d show several modulated results under different scales. However, the results are not satisfactory. With a small ξ, too much details (texture lines) are preserved (Fig. 5b). In contrast, with a larger ξ, the extracted structure lines are blurred or even missing (Fig. 5c&d). One of the main reasons is that the CRF response map simulated by the single-scale Gabor filter fails to provide a good basis for non-CRF modulation. In addition, it lacks of flexibility as it binds the scale of the CRF response simulation to the scale of the texture suppression. Although a recent work proposed to integrate

(a) Input (b) $\xi=1$ (c) $\xi=2$ (d) $\xi=3$ (e) $\xi\in[1,2,3]$

Fig. 5. The modulated results generated by the traditional non-CRF model $\hat{G}_\xi$ with different scales.

the modulated results on different scales [11], the result is only slightly improved (Fig. 5e).

In summary, the traditional models cannot meet our needs in ASCII art synthesis due to the following two reasons:

- The single-scale Gabor filter fails to capture the complete structure of different scales. Even integrating the modulated results of different scales cannot improve the result effectively. Even worse, the integration disables its controllability on the scale of texture suppression.
- The traditional models fail to distinguish weak-contrast structures within a texture region. The weak-contrast structure tends to be mistakenly removed, and results in disappearance of part of the main structure in the synthesized ASCII art.

4.3 Our Non-CRF Modulation

To tackle the above problems, we propose a new model of non-CRF modulation. We first adopt a multi-scale CRF response to capture the structures of different scales in input images. Then we propose a *texture asymmetry* term which can effectively detect and preserve the boundaries of texture region. Fig. 4 shows the flow of our design.

4.3.1 Multi-scale CRF Response Simulation

In order to obtain a complete map of CRF response including structures of different scales, we employ a recently proposed multi-orientation phase congruency model [12] to simulate the CRF response. This multi-orientation phase congruency model can well capture the structures of different scales and preserve weak-contrast structure in natural photographs. For an input image $I(p)$, the CRF response map $F(p)$ is computed as:

$$F(p) = \Big\{ Z_{\theta_j}(p), j \in \{0, 1, \ldots, 5\} \Big\}, \tag{6}$$

where

$$Z_{\theta_j}(p) = \frac{E_{\theta_j}(p)}{\varepsilon + \sum_i A_{\theta_i}(p)}, \tag{7}$$

is the phase congruency value of orientation θ_j. Here,

$$E_{\theta_j}(p) = \sqrt{(\sum_s g^r_{\sigma_s,\theta_j}(p))^2 + (\sum_s g^i_{\sigma_s,\theta_j}(p))^2}, \tag{8}$$

$$A_{\theta_j}(p) = \sum_s \sqrt{g^r_{\sigma_s,\theta_j}(p)^2 + g^i_{\sigma_s,\theta_j}(p)^2}, \tag{9}$$

are the multiscale local energy and amplitude (s is the scale index and we employed 4 scales in our experiments), which are also derived from the Gabor filter defined in Eq. 1, and ε is a small positive constant to avoid the incorrectly highlighting of unnoticeable visual points. According to neurophysiological findings [25], the orientation difference that humans are sensitive to is about $\pi/6$. Therefore, we adopt six orientations that are evenly distributed over range π, i.e., $\theta_j = j\pi/6, j \in \{0, 1, \ldots, 5\}$.

As a result, the CRF response at a pixel $F(p)$ is actually a vector containing six values of phase congruency corresponding to the six selected orientations. With the computed CRF response map, salient structures of different scales can be captured and represented faithfully. In the following, based on the CRF response map, we can adjust the scale of non-CRF modulation. This makes the non-CRF modulation, and hence the subsequent texture suppression, more controllable and flexible.

4.3.2 Non-CRF Modulation Simulation

The traditional non-CRF modulation models blur the boundaries of the texture areas (e.g., the boundaries of the window shutters in Fig. 5d). We propose a new *texture asymmetry* term in our non-CRF modulation model to enhance the preservation of texture boundaries. Our texture asymmetry term is inspired by the asymmetry property in non-CRF suppression among some neurons [7]. The formulation of this term is quite straightforward and efficient. The intuition is that, if the texture features of the two wings (two non-CRF inhibitory regions) are dissimilar for a pixel p, then it is very likely that this pixel is at the boundary of a texture area. Therefore, this pixel should be identified as a structure pixel. We formally define this texture asymmetry term as:

$$Y_{\xi,\theta_j}(p) = 1 - \left(\frac{|S_L(p) - S_R(p)|}{|S_L(p) + S_R(p)|} \right)^n, \tag{10}$$

where $n \in [1, +\infty)$ is a weighting parameter to control the importance of asymmetry during modulation, and

$$S_L(p) = \Psi_l(p; \xi, k, \theta_j) * Z_{\theta_j}(p), \text{and} \tag{11}$$

$$S_R(p) = \Psi_r(p; \xi, k, \theta_j) * Z_{\theta_j}(p), \tag{12}$$

indicate the amount of texture suppression of the two non-CRF inhibitory regions respectively. Recall that $Z_{\theta_j}(p)$ is the CRF response of orientation θ_j defined in Eq. 7, and Ψ_l, Ψ_r are the weighting functions for the two wings defined in Eq. 5.

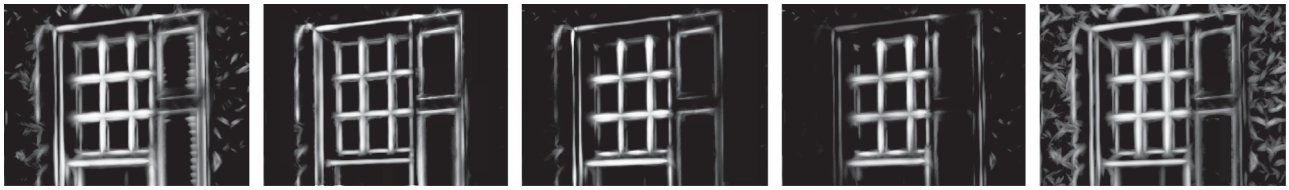
By incorporating the texture asymmetry term, our non-CRF modulation model is defined as:

$$Z'_{\xi,\theta_j}(p) = Z_{\theta_j}(p) - \alpha \cdot Y_{\xi,\theta_j}(p) \cdot \Psi(p; \xi, k, \theta_j) * Z_{\theta_j}(p), \tag{13}$$

where $Z'_{\xi,\theta_j}(p)$ is the modulated result based on the CRF response map $Z_{\theta_j}(p)$. Note that, when the non-CRF regions of p is fully inside a texture region, the textures of the two wings are similar. Therefore, $S_L(p) \approx S_R(p)$, and $Y \approx 1$. In this case, our non- CRF modulation model is similar with the traditional models (Eq. 3). When p is on the boundary of a texture region, the texture difference between the two wings should be larger. Therefore, Y should be much smaller than 1, and the inhibition effect will be largely reduced. To integrate the modulated results from multiple orientations, the orientation with the largest magnitude is chosen:

$$\hat{Z}_\xi(p) = \max_{j \in \{0,1,\cdots,5\}} Z'_{\xi,\theta_j}(p). \tag{14}$$

The weighting parameter n is quite important in our non-CRF modulation model. When n is set to 1, the proposed inhibition term will be determined by $\min\{S_L(p), S_R(p)\}$, which is similar to the idea in [21]. In this case, fine details close to the texture boundary may be mistakenly preserved



(a) $\xi = 2, n = 1.0$ (b) $\xi = 2, n = 1.5$ (c) $\xi = 2, n = 2.0$ (d) $\xi = 2, n = 1000$ (e) $\xi = 1, n = 1.5$

Fig. 6. The modulated results generated by our non-CRF model $\hat{Z}_\xi$. $n = 1.5$ produces the best results empirically.

(e.g., the boundary of the window shutter in Fig. 6a). In contrast, when n is set to a large value, our non-CRF modulation model becomes similar to traditional models. In this case, main structure may be mistakenly suppressed (Fig. 6d). Fig. 6a,b,c,d compare the modulated results when n is set to different values. Increasing n results in a more aggressive suppression. In practice, n is empirically set to 1.5 in our experiments to balance between preserving texture boundary and removing nearby texture.

More importantly, we can easily control the scale of texture suppression by tuning the parameter ξ. Fig. 6b and 6e show two modulation results of two scales where n equals 1.5. Comparing to the traditional non-CRF modulation models (Fig. 5), our model suppresses textures of different scales while preserving the main structure more effectively. Fig. 7 further presents how we control the scale of texture suppression by parameter-tuning. It is observed that the textures on the wings of the butterfly is gradually removed with the increment of texture suppression scale. Note how the structure maps changes as ξ increases, and how they are consistent to the manual ASCII artworks at decreasing text resolution in Fig. 1.

For convenience, we employ a two-tuple $(e(p), o(p))$ to represent the structure map at each pixel, where $e(p) = \hat{Z}_\xi(p)$ and $o(p) = \arg\max_j Z'_{\xi,\theta_j}(p)$ defined in Eq. 14 denote the modulated magnitude and its corresponding orientation respectively. As shown in Fig. 3b, different colors indicate the six orientations, and the luminance indicates the magnitude value.

5 Text Placement

With the extracted structure, our next step is to reproduce the structure map using text characters. To replace structure by characters, the key is to find a text placement so that the text characters can best mimic the structural lines. Since text characters are placed in rows, it is natural to first divide the structure map into a set of horizontal stripes, and then find a row of text characters to reproduce each stripe [12]. For example, in Fig. 8a, the structure map is divided into four horizontal stripes. Four rows of text characters are generated based on the four stripes respectively.

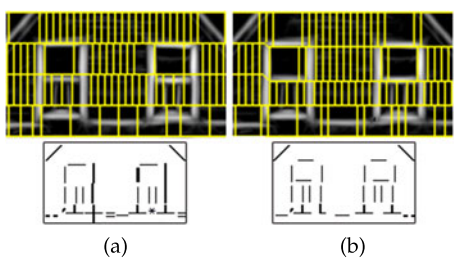
Fig. 7. Three structure maps generated with our method from Fig. 1(a) under different scales of texture suppression. $\xi = 1, 2, 4$, respectively.

However, regularly dividing the structure map into rectangular stripes may not produce satisfactory ASCII art. With a fixed division, the structural lines may locate in places that are difficult to be presented by characters. As a result, important structural lines may be represented by unsuitable characters or even missing in the produced ASCII art (Fig. 8a). In comparison, we found that a much better ASCII art can be produced if the boundary lines between neighboring stripes are slightly shifted upward or downward (Fig. 8b). This suggests that we can divide the structure map into a set of "wavy" horizontal stripes instead of the rectangular ones. Therefore, in order to produce satisfying ASCII art, we propose a novel approach, which flexibly divides the structure map into irregular horizontal stripes so that each horizontal stripe can be well reproduced by a row of text characters.

In order to obtain an optimal stripe division of the structure map, a straightforward idea is to exhaustively test all possible divisions to see if the stripes can be well presented by text. Here, though slight deviation of the dividing boundary may result in better ASCII arts, the deviation should not shift too much from the original position in order to preserve the global structure. In this case, we first add some constrains to the deviation.

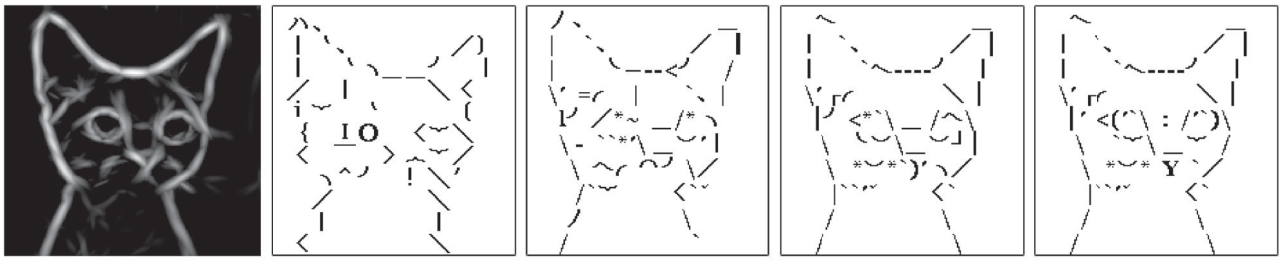
Given a structure map and a user-specified output height h_o (number of text rows), we first evenly divide the structure map into h_o horizontal stripes $I_1, I_2, \ldots, I_{h_o}$. $h_o - 1$ dividing boundaries are formed accordingly between every two neighboring stripes. We denote the dividing boundary between stripes I_k and I_{k+1} as B_k, and the uth pixel on boundary B_k from left to right as $B_k[u]$. With an average division, each boundary B_k is a straight horizontal line. That is, $x(B_k[u+1]) - x(B_k[u]) = 1 \quad \text{and} \quad y(B_k[u]) = y(B_k[u+1])$. Here, $\{x(B_k[u]), y(B_k[u])\}$ is the 2D coordinates of $B_k[u]$. In order to generate better ASCII art, we allow the boundary points to slightly deviate from their original positions vertically. That is,

$$x(\tilde{B}_k[u]) = x(B_k[u]), \tag{15}$$



(a) (b)

Fig. 8. Text placements of (a) fixed stripe division and (b) flexible stripe division.



(a) (b) (c) (d) (e)

Fig. 9. (a) Input image. (b) Text placement with fixed-width characters. (c) Greedy approach with proportional-font characters. (d) Fixed-stripe-boundary text placement with proportional-font characters. (e) Our method.

$$y(\tilde{B}_k[u]) \in [y(B_k[u]) - \epsilon, y(B_k[u]) + \epsilon], \tag{16}$$

where $\tilde{B}_k[u]$ is the deviated boundary point originated from $B_k[u]$. We empirically set ϵ to be 15 percent of the stripe height in all our experiments. Further, the deviated boundary should still be smooth. Therefore, we enforce neighboring pixels on the boundary to be connected with each other, i.e.,

$$|y(\tilde{B}_k[u+1]) - y(\tilde{B}_k[u])| \leq 1. \tag{17}$$

By enforcing the deviation constraints on the dividing boundaries (Eq. 16 & 17), we have already restricted the deviation range of the boundary. But exhaustively testing all potential deviation strategies $\tilde{B}_1, \tilde{B}_2, \ldots, \tilde{B}_{h_o-1}$ is still time-consuming and almost intractable. To boost the efficiency, we sequentially optimize the boundary deviation and text placement stripe-by-stripe from top to bottom. For each stripe I_k, we fix its deviated upper boundary $\tilde{B}_{k-1}$, and optimize its lower boundary $\tilde{B}_k$ and the corresponding text placement T_k. Note that, the deviated upper boundary of the first stripe I_1 is the same of its original upper boundary, i.e., $\tilde{B}_0 = B_0$ where B_0 is the upper boundary of the structure map.

In order to optimize boundary deviation and text placement of a stripe, naive greedy approach of selecting characters one-by-one from left to right does not work. This is because a greedy approach can easily get trapped at a local optimum and hence generate unsatisfactory result (Fig. 9c). In addition, a brute-force search of all possible boundary deviation and text placement is also intractable.

We propose to optimize the boundary deviation and text placement simultaneously via a dynamic programming approach. The idea is that, we divide the boundary deviation and text placement problem of the whole stripe into two subproblems, i.e., replacing the leftmost part of the stripe with a single character, and finding the optimal boundary deviation and text placement for the right part of the stripe. For stripe I_k of width w (pixels), we denote $I_k[u, v]$ as part of I_k from column u to column v. We similarly define $B_k[u, v]$ as part of B_k. $\tilde{I}_k$ is an irregular stripe formed by deviated stripe boundaries $\tilde{B}_{k-1}$ and $\tilde{B}_k$. Then, our problem of finding the best deviated boundary $\tilde{B}_k$ and the corresponding text placement T_k for I_k can be divided into two subproblems, i.e., finding the best deviated boundary $\tilde{B}_k[1, u]$ and the corresponding character c to replace $\tilde{I}_k[1, u]$, and finding the best deviated boundary $\tilde{B}_k[u+1, w]$ and the corresponding character sequence T'_k to replace $\tilde{I}_k[u+1, w]$, which is formulated as:

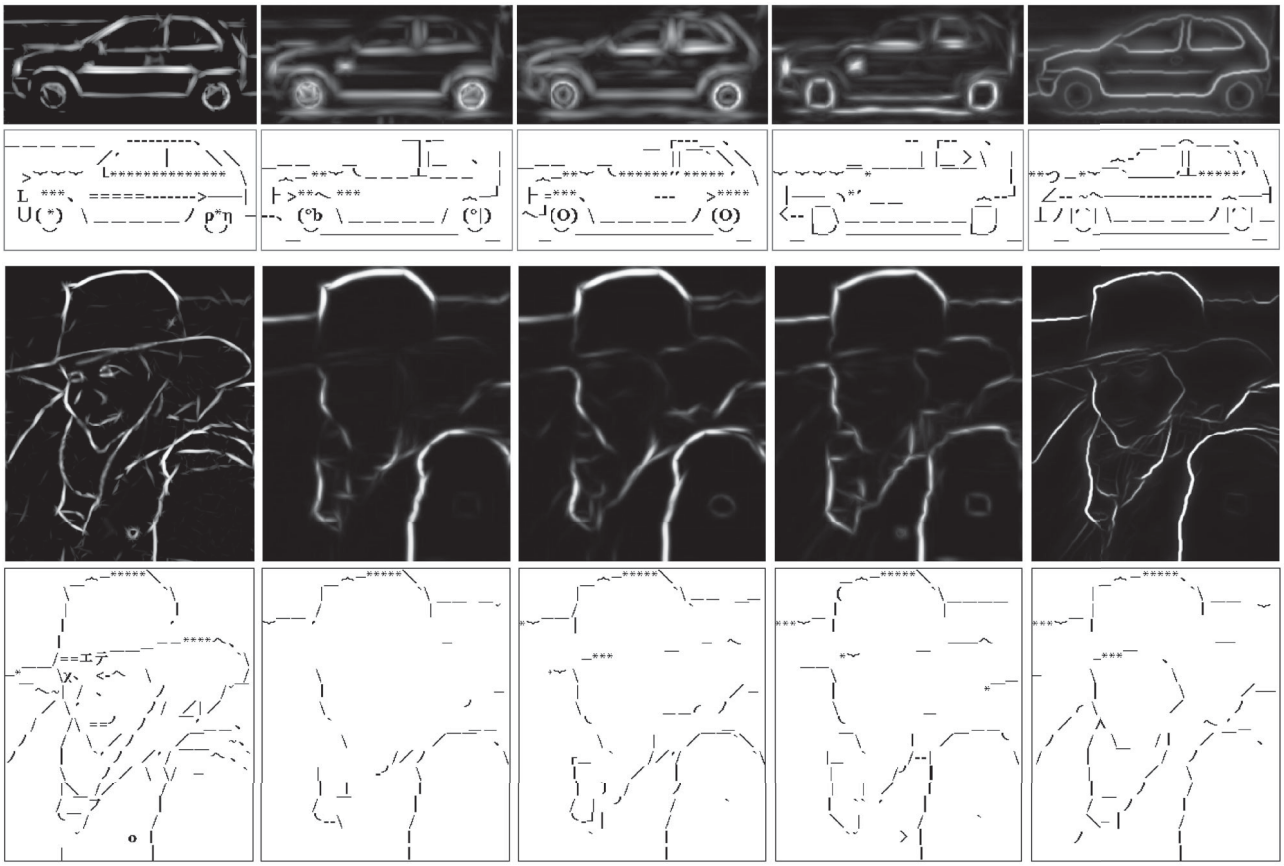
$$\begin{aligned}\{T_k, \tilde{B}_k\} &= \arg\max_{T_k, \tilde{B}_k} D_{SM}\big(R(\tilde{I}_k), I_S(T_k)\big) \\ &= \arg\max_{u, T'_k, c, \tilde{B}_k} D_{SM}\big(R(\tilde{I}_k[1,u]), I_S(c)\big) \\ &\quad + D_{SM}\big(R(\tilde{I}_k[u+1,w]), I_S(T'_k)\big).\end{aligned} \tag{18}$$

Here, all potential candidate character sequences T_k will be evaluated under all possible boundary variations $\tilde{B}_k$ suggested by Eq. 15, 16, 17. Since the stripe boundary may deviate from the original position, the deviated image stripe may become irregular. To correspond a character sequence to an irregular stripe $\tilde{I}_k$, we adopt a rectification function $R(\tilde{I}_k)$ [26] to rectify the irregular stripe $\tilde{I}_k$ into the original rectangular shape. Characters are also resized to fit the height of the rectified stripe. $I_S(T_k)$ is a function which generates a structure image for a character sequence T_k. Furthermore, $D_{SM}(S, S')$ is an image similarity metric for measuring the similarity of two structural images S and S' obtained based on Eq. 14. To evaluate the similarity of two structural images, we employ the *polar distance* metric. In addition, a *point-to-area* matching mode [27] is adopted for tolerating deformation. We define it as:

$$\begin{aligned}&D_{SM}(S, S') = (D(S, S') + D(S', S))/2 \\ &D(S, S') \\ &= \sum_{p \in S} \min_{q \in S' \cap \Lambda(p)} \sqrt{(e(p))^2 + (e'(q))^2 - 2e(p)e'(q)\cos(o(p) - o'(q))},\end{aligned} \tag{19}$$

where $e(p)$ and $o(p)$ are the magnitude and orientation, respectively, of pixel p in the structure map. $\Lambda(p)$ indicates the neighborhood of p defined by $\forall q, ||p - q|| < r$. Parameter r is the degree of deformation tolerance and is set to 5 in all our experiments. Note that this similarity measurement is bidirectional in order to achieve reliability.

With this formulation, we can easily observe that Eq. 18 is in fact a standard formulation which can be solved via a dynamic programming approach. Here, the boundary pixels, together with their deviations, form a 2-D grid for accumulating character matching error. The constraint defined in Eq. 17 is accounted when accumulating the matching error. Our dynamic programming formulation can be solved either in a left-to-right or right-to-left way, which may lead to different character matching results. If the



(a) Our model (b) Xu et al. [2] (c) Zhang et al. [3] (d) Cho et al. [4] (e) gPb [5]

Fig. 10. "Car" and "Woman". The first and third rows are the structure maps. The second and fourth rows are the corresponding ASCII arts. Note that our method preserves the contour of the car and the woman's face while removing the texture in the background and the scarf of the woman.

per-stripe optimization is performed in a left-to-right way, there may leave a small residue in the rightmost area; if the optimization is performed in a right-to-left way, there may leave a small residue in the leftmost area. Since it is natural to place letters in a left-to-right way, we solve the dynamic programming formulation in a left-to-right way, which may lead to a small residue left in the rightmost part. Note that, the optimal boundary deviation and text placement for each stripe are simultaneously determined in the process. Once the optimal boundary deviation is determined, the optimal text placement is also determined. One may also notice that when we optimize the deviation of the lowest division boundary B_{h_o-1}, we need to take both neighboring stripes I_{h_o-1} and I_{h_o} into consideration in order to generate satisfying ASCII art results. Fig. 9e visualizes an ASCII art generated from our method. Comparing with Xu's method [12] (Fig. 9d), our method allows the dividing boundaries to slightly shift upward or downward from the original positions. Therefore, the generated ASCII art result has more freedom corresponding to the flexibility of stripe deformation, which results in smoother and visually more pleasant ASCII art results.

6 Results and Discussions

To validate the effectiveness of our method, we first compare our structure extraction method with the state-of-the-art texture filters and contour detectors, including Xu et al. [2], Zhang et al. [3], Cho et al. [4], and gPb [5]. Then, we compare our ASCII art generation method with the state-of-the-art automatic ASCII art generation methods [1], [12].

6.1 Structure Extraction Comparison

Fig. 10 compares our structure extraction method to the existing texture filtering and contour detection methods. The first and third rows present the extracted structure maps, while the second and fourth rows present the ASCII arts generated from the corresponding structure maps. The original photographs are shown in Fig. 11a. Note that the texture filters are not originally designed for extracting structure (Fig. 10b,c,d), but their methods can be extended to extract structure by applying multi-scale Gabor filter on the smoothed images (Fig. 11).

In the "car" example of Fig. 10, the background contains rich texture (the fences and the checked floor). If these excessive amount of texture are preserved, the generated ASCII art (especially for low text resolution ones) would be overcrowded. In order to smooth these background texture, existing texture filtering methods undesirably blur the contour of the car as well (Fig. 10b, 10c, and 10d). While the contour detector gPb can somehow maintain the texture boundary, it suffers from the structure deformation (Fig. 10e). In contrast, our method not only extracts the contour of the car effectively, but also suppress the background texture (Fig. 10a). In the "woman" example of Fig. 10, the scarf also contains busy texture. In order to remove both the texture on the scarf and the

(a) Original image (b) Xu et al. [2] (c) Zhang et al. [3] (d) Cho et al. [4]

Fig. 11. "Car" and "Women". The original photographs and the smoothed results of the texture filtering methods.

irregular texture in the background, both the texture filters and the contour detector simultaneously undesirably blur the face (Figs. 10b, 10c, 10d). Note that with a less aggressive parameter tuning, these methods are able to preserve the face structure, with a trade-off of retaining the background and scarf textures, and fails to produce concise ASCII art. In sharp contrast, our method successfully suppress the texture while preserving important face structure (Fig. 10a), and hence a concise ASCII art that faithfully reproduces the input photograph.

6.2 ASCII Art Comparison

6.2.1 Visual Comparison

Our goal is to generate ASCII art directly from natural photographs. Currently, the only work that accepts natural photographs as input is Xu et al. [12], but their method cannot control the scale of the extracted structures. Another automatic structure-based ASCII art generation method is proposed by Xu et al. [1] which only accepts line drawings as input. To isolate the effect of structure extraction, we use the structure map generated by our structure extraction method to feed into the method of Xu et al. [1]. In addition, we ask the artists to create ASCII arts based on the input photographs. Note that both Xu et al.[12] and our method can generate ASCII art with both proportional and fixed-width fonts, while Xu et al.[1] can only handle fixed-width fonts.

Traditional ASCII art only considers *ASCII-only* characters (totally 95 characters), while modern ASCII art utilizes also a richer character set in order to present more complex structure. Therefore, in our work, our character set is extended to contain 305 characters, which has been listed in supplementary material, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TVCG.2016.2569084. In this paper we use the typeface of *Geogia* for all the experiments, although our system can accept any typeface.

Fig. 12 compares the ASCII arts generated by Xu et al. [1], Xu et al.[12], our method with ASCII-only set, our method with our extended set, and the artist work. Note that the structure maps are automatically obtained according to the target text resolutions (all the structure maps can be found in the supplementary materials, available online). It is observed that the ASCII arts generated via Xu et al. [1] (second row) are visually unsatisfactory even though they use the same structure map. This is mainly due to the rigidity of fixed-width characters and their placement. Though the ASCII arts generated by Xu et al. [12] (third row) use proportional fonts, the results are overcrowded as they do not suppress unwanted texture. In comparison, our results (fourth and fifth rows) are more concise and faithfully reproduce the visual content in the input, thank to our controllable texture suppression and weak-contrast structure preservation. Note how the grid on the tennis racket (left column), the grid on both the hat and the T-shirt (middle column), and the line beams on the wheels (right column) are effectively suppressed. Moreover, compared to the results with ASCII-only character set, those with extended character set are more visually appealing since the extended characters can present more complex structure. Nevertheless, the creations of the artist (last row) are the most ingenious and interesting as they can intelligently deform and even modify the content for creating ASCII arts. For example, although the grid of tennis racket is too fine to reproduce under the target text resolution, the artist can ingeniously represent them with a group of letters "x".

Fig. 13 shows our results for three different target text resolutions. For each target text resolution, our method generates the corresponding structure map by adjusting the texture suppression scale. Note that how the scale of texture suppression is increased as the target text resolution decreases. Such controllability of the texture suppression scale allows us to suppress crowded texture so as to effectively fit only the main structure into the desired display resolutions. Fig. 14 presents one more result where the input is a complicated natural photograph with a wide range of textures. The result demonstrates the effectiveness of our method in preserving main structure for ASCII art generation while suppressing the overwhelming texture.

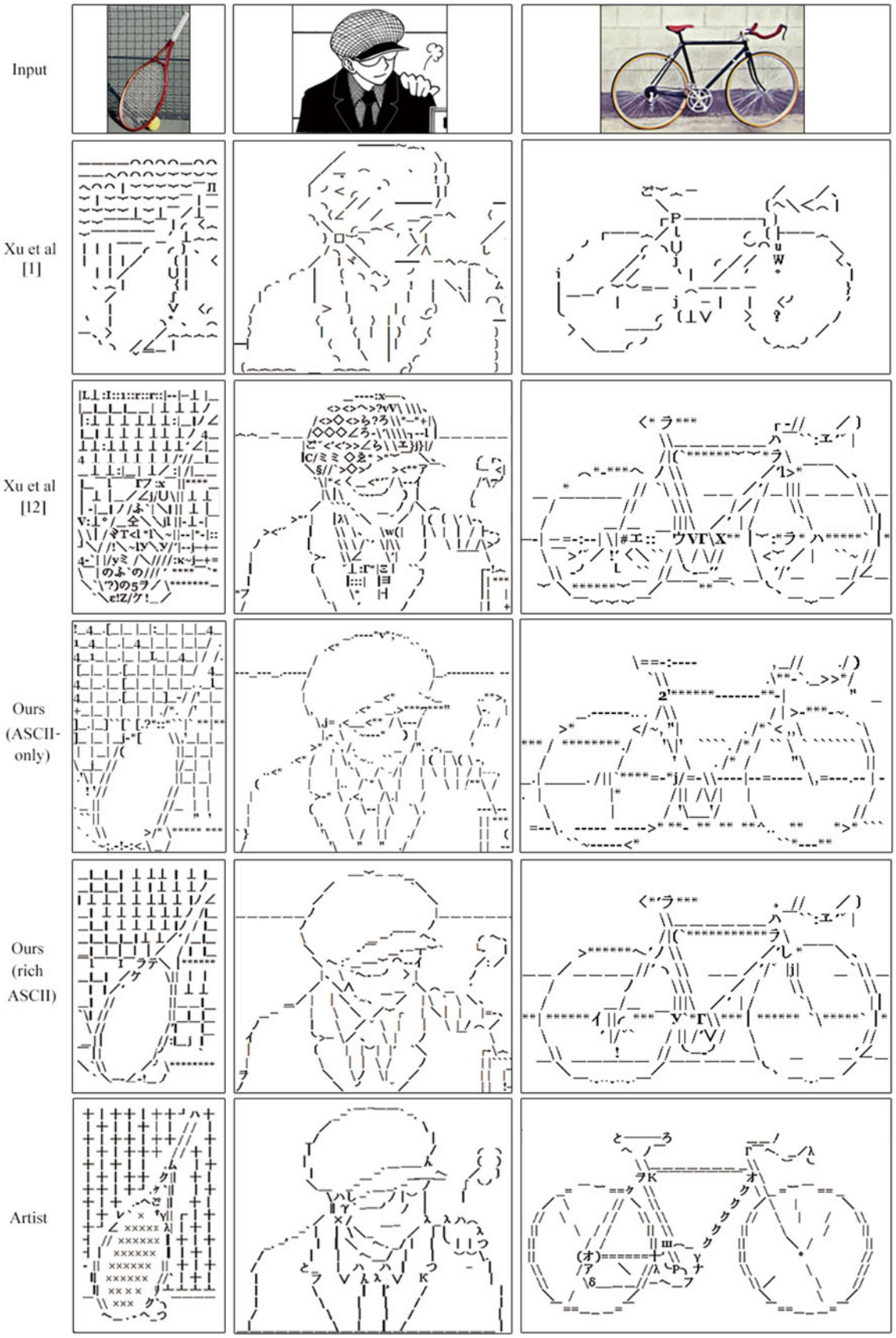


Fig. 12. Comparisons with the existing ASCII art generation methods and the artist's creation.

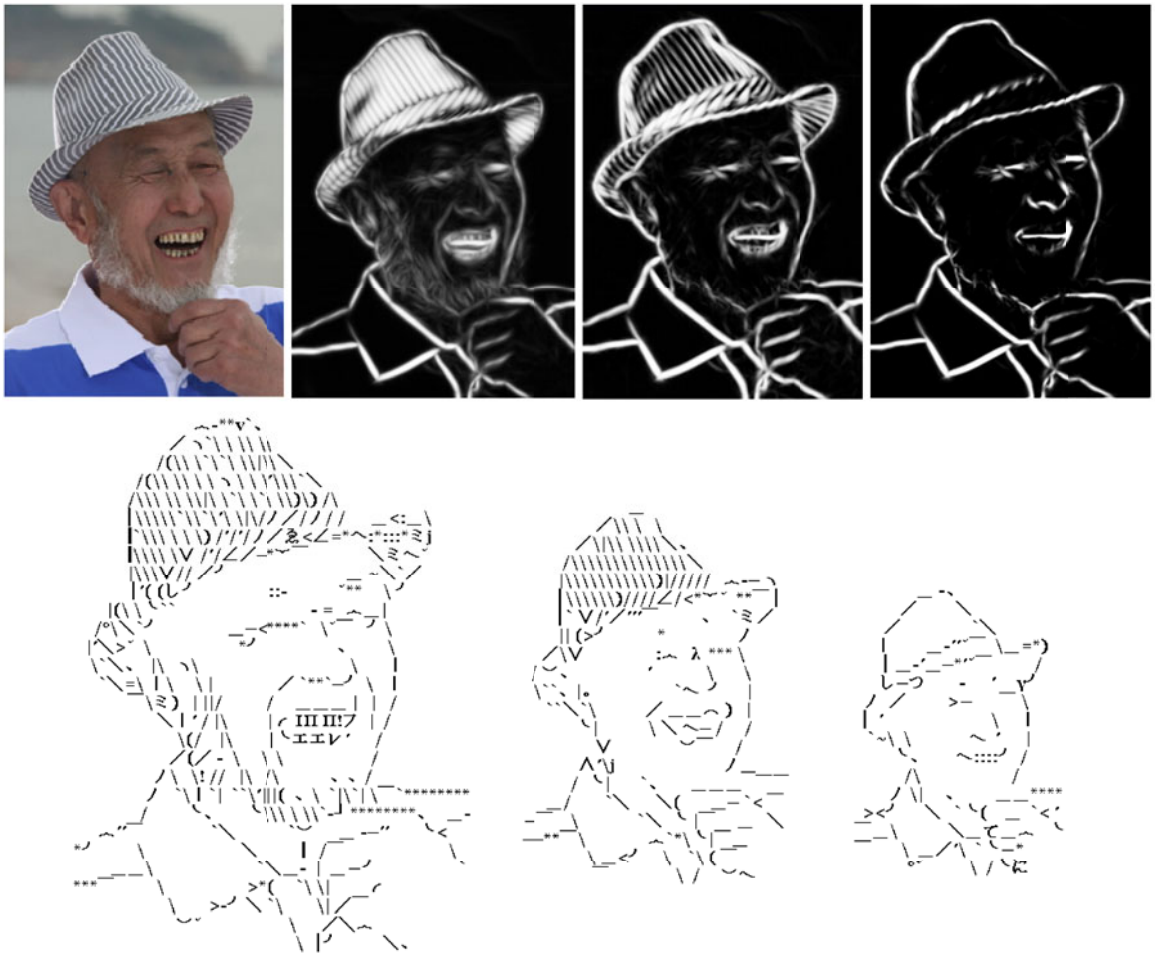


Fig. 13. "Old man". By controlling the scale of texture suppression, we can generate structure maps with different amount of structures and create ASCII arts for different resolutions.

6.2.2 User study

We further conducted a user study to evaluate our system. We select nine natural photographs and five cartoons with different contents and styles. For each photograph, we prepare three ASCII arts generated by our method, Xu et al. [1] and Xu et al. [12] respectively. We also invited the artists to create the ASCII arts for seven photographs. The reason that only seven artworks are created by the artist is due to the tediousness and complexity in manual creation. To minimize the learning effects during the user study, the generated ASCII arts are shown in different ordering to different participant. The input natural photographs are also presented aside the ASCII arts for reference. The users are asked to rate each ASCII art from 1 to 10 in terms of recognition ability, ingeniousness, and aesthetics. Fifty participants are invited. Among them, 23 are males and 27 are females, and their ages range from 16 to 72. The test cases and details of the design of user study can be found in the supplementary materials, available online.

Table 1 shows the average scores of different terms comparing our method with the other methods. Regarding the average scores, Xu et al. [12] is slightly lower than ours, because it reproduces the excessive amount of texture and results is overcrowded. Xu et al. [1] is the worst since the fixed-width font cannot well represent a variety of complex structure. Comparing to manual artworks, our method is less ingenious since artists can modify or even drop the content to facilitate the ASCII art composition. Table 2 shows the analysis results using p values computed by t-test. Generally, $p < 0.05$ indicates the difference of two data is reliable. Thus, p values shown in Table 2 illustrate that the average scores in Table 1 can reliably reflect the difference of our method and Xu et al. [1] in all three terms, and Xu et al. [12] in terms of ingeniousness. On the other hand, the difference between our results and the artist's works is not significant. This implies that our method can clearly improve the visual effect of ASCII art compared to Xu et al. [1], and generates comparable results compared to artists.

Timing Performance. Our experiments are conducted on a PC with 2 GHz CPU, 8 GB memory. No GPU is utilized during the computation. For a $1{,}024 \times 1{,}024$ input image, it takes about 2 seconds for structure extraction, and about 18 minutes for the text placement. Our time complexity is about $O(n^2)$, where n is the number of pixels. Note that the resolutions of most images presented in the paper and the supplementary material, available online, are or lower than $1{,}024 \times 1{,}024$.

Limitations. Currently, our structure extraction does not "repair" the broken structure. We may explore the way to re-connect broken structure during the structure extraction for ASCII art generation. Currently, our method does not

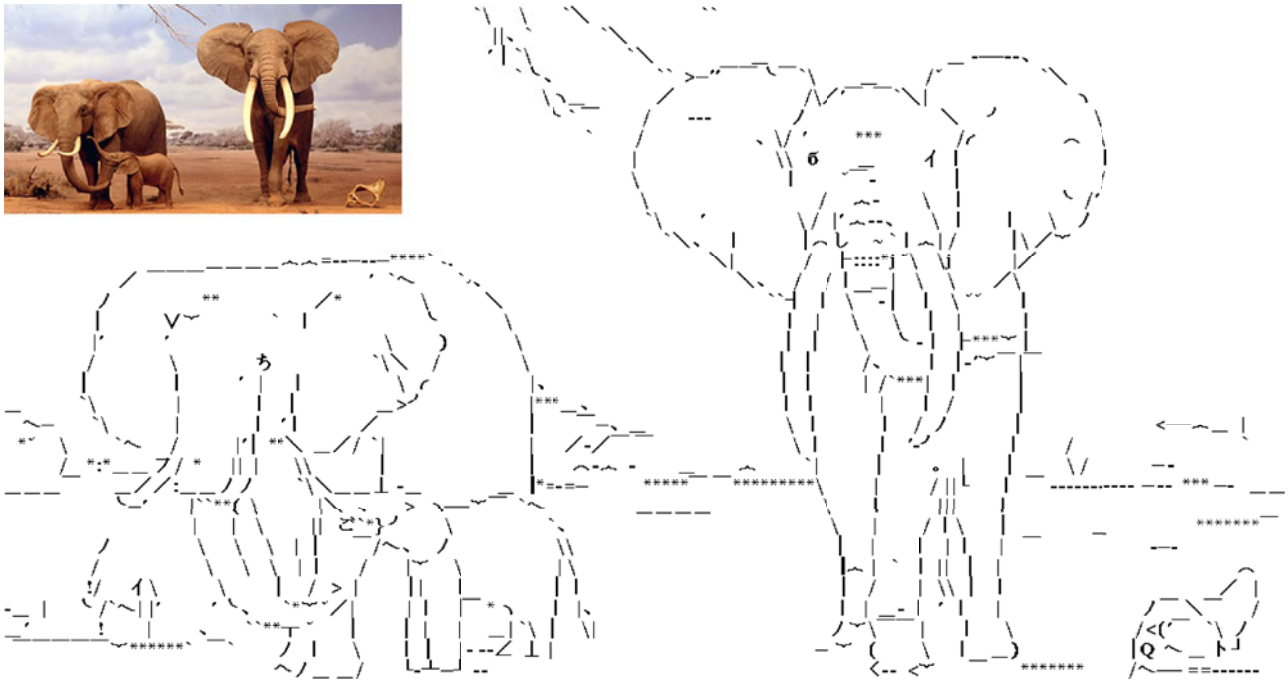


Fig. 14. "Elephant". The input is a complex photo. Our method successfully generates a visually pleasing ASCII art.

TABLE 1
User Study Statistics Using Average Score and Standard Deviation. In Each Entry Above, n_1/n_2 Indicate the Average Score and Standard Deviation Respectively

	Recognition ability	Ingeniousness	Aesthetics
Ours	8.93 / 1.23	8.63 / 0.97	8.57 / 0.57
Xu et al. [1]	6.31 / 1.27	5.97 / 1.09	5.58 / 1.31
Xu et al. [12]	8.57 / 1.31	7.88 / 1.01	8.35 / 0.83
Artist	8.82 / 1.31	9.03 / 0.93	8.62 / 0.53

TABLE 2
User Study Statistics Using p Value

	Recognition ability	Ingeniousness	Aesthetics
Ours	-	-	-
Xu. et al. [1]	7.69E-10	1.88E-12	1.78E-15
Xu. et al. [12]	0.1606	0.005	0.1360
Artist	0.3804	0.9285	0.6254

take color information into consideration during ASCII art generation. While color text is available on most instant messaging platforms, incorporating color information into ASCII art generation may further improve the visual impact.

7 Conclusion

In this paper, we proposed method to generate structure-based ASCII art directly from the complex natural photographs. As the natural photographs may contain excessive amount of texture and unclear structure, existing ASCII art generation methods usually fail to produce visually pleasant ASCII arts. Based on the physiological findings, we design our own model of the nCRF modulation to preserve weak-contrast structure while suppressing undesired texture. Users can optionally control the scale of texture to suppress, or simply leave it to the system to control the texture suppression. The proposed structure extraction can also be applicable in other pattern recognition problems. To optimally place the proportional fonts, we formulate the text placement as a dynamic programming problem. Bothy qualitative and quantitative experiments are conducted and results show that our method outperforms the existing methods in various aspects.

Acknowledgment

This work was supported by the funding from the NSFC (Grant No. 61272293, 61300137, 61472145, 61233012) and NSFG (Grant No. S2013010014973), RGC Fund (Grant No. CUHK14200915), Science and Technology Planning Major Project of Guangdong Province (Grant No. 2015A070711001), Open Project Program of Guangdong Key Lab of Popular High Performance Computers and Shenzhen Key Lab of Service Computing and Applications (Grant No.SZU-GDPHPCL2015).

References

[1] X. Xu, L. Zhang, and T. Wong, "Structure-based ASCII art," *ACM Trans. Graph.*, vol. 29, no. 4, pp. 52:1–52:9, 2010. [Online]. Available: http://doi.acm.org/10.1145/1833351.1778789

[2] L. Xu, Q. Yan, Y. Xia, and J. Jia, "Structure extraction from texture via relative total variation," *ACM Trans. Graph.*, vol. 31, no. 6, p. 139, 2012. [Online]. Available: http://doi.acm.org/10.1145/2366145.2366158

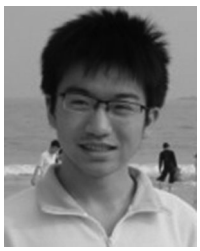
[3] Q. Zhang, X. Shen, L. Xu, and J. Jia, "Rolling guidance filter," in *Proc. Eur. Conf. Comput. Vision*, 2014, pp. 815–830.

[4] H. Cho, H. Lee, H. Kang, and S. Lee, "Bilateral texture filtering," *ACM Trans. Graph.*, vol. 33, no. 4, p. 128, 2014. [Online]. Available: http://doi.acm.org/10.1145/2601097.2601188

[5] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, May 2011. [Online]. Available: http://doi.ieeecomputersociety.org/10.1109/TPAMI.2010.161

[6] P. Dollár and C. L. Zitnick, "Structured forests for fast edge detection," in *Proc. IEEE Int. Conf. Comput. Vision*, 2013, pp. 1841–1848. [Online]. Available: http://dx.doi.org/10.1109/ICCV.2013.231
[7] H. E. Jones, K. L. Grieve, W. Wang, and A. M. Sillito, "Surround suppression in primate v1," *J. Neurophysiol.*, vol. 86, no. 4, pp. 2011–2028, 2001.
[8] C. Grigorescu, N. Petkov, and M. A. Westenberg, "Contour detection based on nonclassical receptive field inhibition," *IEEE Trans. Image Process.*, vol. 12, no. 7, pp. 729–739, Jul. 2003. [Online]. Available: http://dx.doi.org/10.1109/TIP.2003.814250
[9] G. Papari and N. Petkov, "An improved model for surround suppression by steerable filters and multilevel inhibition with application to contour detection," *Pattern Recognition*, vol. 44, no. 9, pp. 1999–2007, 2011.
[10] C. Zeng, Y. Li, K. Yang, and C. Li, "Contour detection based on a non-classical receptive field model with butterfly-shaped inhibition subregions," *Neurocomputing*, vol. 74, no. 10, pp. 1527–1534, 2011.
[11] H. Wei, B. Lang, and Q. Zuo, "Contour detection model with multi-scale integration based on non-classical receptive field," *Neurocomputing*, vol. 103, pp. 247–262, 2013. [Online]. Available: http://dx.doi.org/10.1016/j.neucom.2012.09.027
[12] X. Xu, L. Zhong, M. Xie, J. Qin, Y. Chen, Q. Jin, T.-T. Wong, and G. Han, "Texture-aware ASCII art synthesis with proportional fonts," in *Proc. 5th Int. Symp. Non-Photorealistic Animation Rendering*, 2015, PP. 183–193.
[13] P. D. O'Grady and S. T. a. Rickard, "Automatic ASCII art conversion of binary images using non-negative constraints," in *Proc. IET Irish Signals Syst. Conf., 208*, 2008, pp. 186–191.
[14] K. Miyake, H. Johan, and T. Nishita, "An interactive system for structure-based ASCII art creation," *Proc. NICOGRAPH Int.*,2011, pp. 4–3.
[15] M. Krzywinski, "ASCII art proportional spacing, tone/structure mapping and fixed strings," 2011 [Online]. Available: http://mkweb.bcgsc.ca/asciiart/
[16] N. Markuš, M. Fratarcangeli, I. S. Pandžić, and J. Ahlberg, "Fast rendering of image mosaics and ascii art," *Comput. Graph. Forum*, vol. 34, no. 6, pp. 251–261, 2015.
[17] H. Kang, S. Lee, and C. K. Chui, "Coherent line drawing," in *Proc. 5th Int. Symp. Non-Photorealistic Animation Rendering*, 2007, pp. 43–50. [Online]. Available: http://doi.acm.org/10.1145/1274871.1274878
[18] L. Karacan, E. Erdem, and A. Erdem, "Structure-preserving image smoothing via region covariances," *ACM Trans. Graph.*, vol. 32, no. 6, pp. 176:1–176:11, Nov. 2013. [Online]. Available: http://doi.acm.org/10.1145/2508363.2508403
[19] M. Aubry, S. Paris, S. W. Hasinoff, J. Kautz, and F. Durand, "Fast local laplacian filters: Theory and applications," *ACM Trans. Graph.*, vol. 33, no. 5, pp. 167:1–167:14, Sep. 2014. [Online]. Available: http://doi.acm.org/10.1145/2629645
[20] Z. Shen, W. Xu, and C. Li, "Cue-invariant detection of centre–surround discontinuity by v1 neurons in awake macaque monkey," *J. Physiol.*, vol. 583, no. 2, pp. 581–592, 2007.
[21] C. Zeng, Y. Li, K. Yang, and C. Li, "Contour detection based on a non-classical receptive field model with butterfly-shaped inhibition subregions," *Neurocomputing*, vol. 74, no. 10, pp. 1527–1534, 2011. [Online]. Available: http://dx.doi.org/10.1016/j.neucom.2010.12.022
[22] K. Yang, C. Li, and Y. Li, "Multifeature-based surround inhibition improves contour detection in natural images," *IEEE Trans. Image Process.*, vol. 23, no. 12, pp. 5020–5032, Dec. 2014. [Online]. Available: http://dx.doi.org/10.1109/TIP.2014.2361210
[23] Q. Tang, N. Sang, and T. Zhang, "Extraction of salient contours from cluttered scenes," *Pattern Recognition*, vol. 40, no. 11, pp. 3100–3109, 2007. [Online]. Available: http://dx.doi.org/10.1016/j.patcog.2007.02.009
[24] C. Grigorescu, N. Petkov, and M. A. Westenberg, "Contour detection based on nonclassical receptive field inhibition," *IEEE Trans. Image Process.*, vol. 12, no. 7, pp. 729–739, Jul. 2003. [Online]. Available: http://dx.doi.org/10.1109/TIP.2003.814250
[25] T. S. Lee, "Image representation using 2D Gabor wavelets," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 10, pp. 959–971, Oct. 1996. [Online]. Available: http://doi.ieeecomputersociety.org/10.1109/34.541406
[26] Bookstein and F. F.L., "Principal warps: Thin-plate splines and the decomposition of deformations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 6, pp. 567–585, Jun. 1989.
[27] W. Jia, D. Huang, and D. Zhang, "Palmprint verification based on robust line orientation code," *Pattern Recognition*, vol. 41, no. 5, pp. 1504–1513, 2008. [Online]. Available: http://dx.doi.org/10.1016/j.patcog.2007.10.011

Xuemiao Xu received the BS and MS degrees from the South China University of Technology and PhD degree from the Chinese University of Hong Kong in 2002, 2005 and 2009, respectively. She is currently a professor in School of Computer Science and Engineering, South China University of Technology. Her research interests include computer graphics, non-photorealistic rendering, and intelligent transportation based on image analysis.



Linyuan Zhong received the BS degree in computer science and engineering from South China University of Technology in 2013. And he is currently working toward the MS degree at the Department of Computer Science and Engineering, South China University of Technology. His research interests are computer graphics, computer vision, and non-photorealistic rendering.

Minshan Xie received the BS degree in software engineering from South China University of Technology in 2015. And she is currently working toward the MS degree at the Department of Computer Science and Engineering, South China University of Technology. Her research interests are computer graphics, computer vision, and non-photorealistic rendering.

Xueting Liu received the BEng degree from Tsinghua University and PhD degree from the Chinese University of Hong Kong in 2009 and 2014, respectively. She is currently a postdoctoral research fellow at the Department of Computer Science and Engineering, the Chinese University of Hong Kong. Her research interests include computer graphics, computer vision, computational manga and anime, and non-photorealistic rendering.

Jing Qin received the PhD degree in computer science and engineering from the Chinese University of Hong Kong. He is currently an assistant professor at the Centre for Smart Health, School of Nursing, the Hong Kong Polytechnic University. His research interests include physically based modeling, visualization, medical image processing, computer-assisted surgery and computer graphics.

Tien-Tsin Wong received the BSc, MPhil, and PhD degrees in computer science from the the Chinese University of Hong Kong in 1992, 1994, and 1998, respectively. He is currently a professor at the Department of Computer Science and Engineering, the Chinese University of Hong Kong. His main research interests include computer graphics, computational manga, precomputed lighting, image-based rendering, GPU techniques, multimedia compression, and computer vision. He received the IEEE Transactions on Multimedia Prize Paper Award 2005 and the Young Researcher Award 2004.