Vis Comput (2014) 30:969–979
DOI 10.1007/s00371-013-0881-6

ORIGINAL ARTICLE

Artistic preprocessing for painterly rendering and image stylization

Yu Zang · Hua Huang · Chen-Feng Li

Published online: 27 November 2013

Abstract A practical image enhancing technique is presented as a preprocessing step for painterly rendering and image stylization, which transforms the input image mimicking the vision of artists. The new method contains mainly two parts dealing with artistic enhancement and color adjustment, respectively. First, through feature extraction and simplification, an abstract shadow map is constructed for the input image, which is then taken as a guide for emphasizing the light–shadow contrast and the important shadow lines. Next, to simulate the intense color emotion often subjectively added by the artist, a color adjustment technique is proposed to generate lively colors with sharp contrast imitating the artistic vision. The preprocessing operation is compatible with existing stylization and stroke-based painterly rendering techniques, and it can also produce different types of stylization results independently.

Keywords Image processing · Example based painting · Color features learning

Y. Zang
School of Electrical Information Engineering, Xi'an Jiaotong University, Shaanxi, China
e-mail: zangyu7@126.com

H. Huang (✉)
Beijing Key Laboratory of Intelligent Information Technology, School of Computer Science and Technology, Beijing Institute of Technology, Beijing, China
e-mail: huanghua_xjtu@hotmail.com

H. Huang
e-mail: huahuang@bit.edu.cn

C.-F. Li
College of Engineering, Swansea University, Wales, UK
e-mail: c.f.li@swansea.ac.uk

1 Introduction

Non-photorealistic rendering (NPR) techniques aim to create painting-like results from real photos/videos. Current NPR techniques focus mainly on simulating artists' actual painting process, in order to "leave the mechanical details of artistic creation to a computer." Increasingly more sophisticated rendering systems have been developed to simulate more closely "how artists paint," but little attention has been paid on "how artists observe." As shown in Fig. 1, real paintings often feature strong light–shadow contrast, important lines emphasis and lively color scheme. It is the combination of these artistic features that delivers a vibrant and strong visual impact. However, NPR results are often seen to be strongly dependent on the quality of input photos, and the result can appear flat and blunted if the input image is short of artistic features. To overcome this problem, we focus on the "observation" side of art creation, and aim to enhance the input image realizing artistic vision so that it can be used for a better rendering effect.

By comparing real paintings of different styles, we identify three key features that reflect the artistic vision. First, it is noticed that the relationship between light and shadow is often simplified, and the shadow area is usually drawn on a separate layer to enhance the contrast between light and dark, especially for cartoon works (see Fig. 1(a)). Secondly, the line draft is employed not only to describe the overall composition of an artwork, but also to emphasize some important shadow lines (see Fig. 1(a–b)). Last but not least, the color scheme tends to be exaggerated and contrasted to produce stronger visual impact (see Fig. 1(a–b)).

Based on the above observation, we present an image enhancement technique to mimic artistic observation, which includes two main operation steps. In the artistic enhancement step, a simplified shadow layer is extracted from the

(a) (b)

Fig. 1 Some real paintings of different styles. Photo credit: Flickr user Ahmed (**a**) and wizan (**b**)

input image, which is then used for the enhancement of light–shadow contrast. Based on the abstract shadow map, a Gabor feature behavior analysis technique is proposed to identify and emphasize the significant contours and shadow lines at different levels. In the color adjustment step, a modulating scheme adapted to real toning process is developed to create lively colors with sharp contrast. Experimental results show that our preprocessing technique can add lively artistic features to a wide range of input images, and, as a supplementary tool, it can be readily connected to existing stroke-based rendering systems to produce better artistic effect.

2 Related work

Over the past decade, stroke-based painterly rendering techniques have been extensively studied [11, 20, 22]. By considering temporal coherence in video clips, Litwinowicz [24] designed an automatic filter to produce impressionism animations in a hand-drawing style. Hertzmann [9] proposed a multi-level stroke distribution strategy with a curved stroke model, in which the painting style can be adjusted by some tuning parameters. Hays and Essa [8] proposed an RBF-interpolation based technique to compute the stroke orientation, which works particularly well for images with large smooth content areas. Zeng et al. [42] presented a new rendering framework based on a hierarchial parsing tree to describe the semantic information of an image, and their approach can generate oil-painting effects similar to human perception. Based on the same idea, Zhao and Zhu [45] presented an interactive painting system "Sisley" which can help amateur users to create their own artistic works. Focused mainly on video, Lee et al. [21] proposed a novel technique to compute the stroke orientation field by taking into account object movements in the input video clip. By using a segmentation based rendering technique with a new stroke assignment strategy, Kagaya et al. [16] developed an improved framework to generate rendering results in multiple styles. To further improve temporal coherence in video rendering, Huang et al. [13] proposed a video layering based painterly rendering technique to combine layering and painting techniques, such that flickering artifacts are reduced considerably. O'Donovan and Hertzmann [28] proposed an interactive painting system which allows more direct operation of the paint creation by supporting spectrum interaction, and demonstrated a new idea to control the overall painting process.

The methods discussed above focus mainly on the rendering side and pay less attention to enhancing the input image itself, so the rendering results can potentially be unsatisfactory especially when the input images are lack of artistic features.

Some image abstraction and stylization research also provides inspiration for this work. DeCarlo and Santella [6] employed the mean-shift segmentation [4] to guide the simplification of image content in a hierarchical system, and a human perception model was used along with a set of eye tracking data to identify the significant structures. Wang et al. [35] and Collomosse et al. [3] extended this work into 3-dimensional space. Specifically, Wang et al. [35] treated the video as a space-time volume and proposed an anisotropic kernel mean shift technique to generate cartoon-like results, and Collomosse et al. [3] proposed a similar mean-shift based system which not only maintained good spatio-temporal coherence but was also able to produce more artistic styles, such as cartoon, sketchy, oil and watercolor paintings. Wen et al. [37] presented a color sketch generation system, based on an interactive mean-shift algorithm equipped with a boundary shrinking and color shift strategy, and the system could produce freehand drawing styles. Winnemöller et al. [38] modified the mean-shift based abstraction framework by employing the bilateral filter to smooth the input media data and using the Difference of Gauss (DoG) edge model to further increase the distinction of high contrast regions. This technique could produce cartoon-like effects with good computational efficiency. Kang et al. [18] further improved this technique by introducing a flow-based image abstraction technique. In their system, the line extraction and the smoothing process were guided by a vector field which describes the flow of salient features, and the quality of results was improved considerably. Zhang et al. [44] proposed an online video stream stylization technique which can maintain spatio-temporal coherence, and by using an additional color template, the color style of the results was in accordance with typical cartoon works. Kyprianidis and Kang [19] proposed an image/video stylization framework which is based on a flow field guided adaptive line integral convolution and shock filtering, and this approach can effectively generate the abstraction version of the input while preserving the overall image structure.

The above abstraction and stylization techniques deemphasize "insignificant" image features, while paying little attention to artwork-like representation. Although Wen et al.

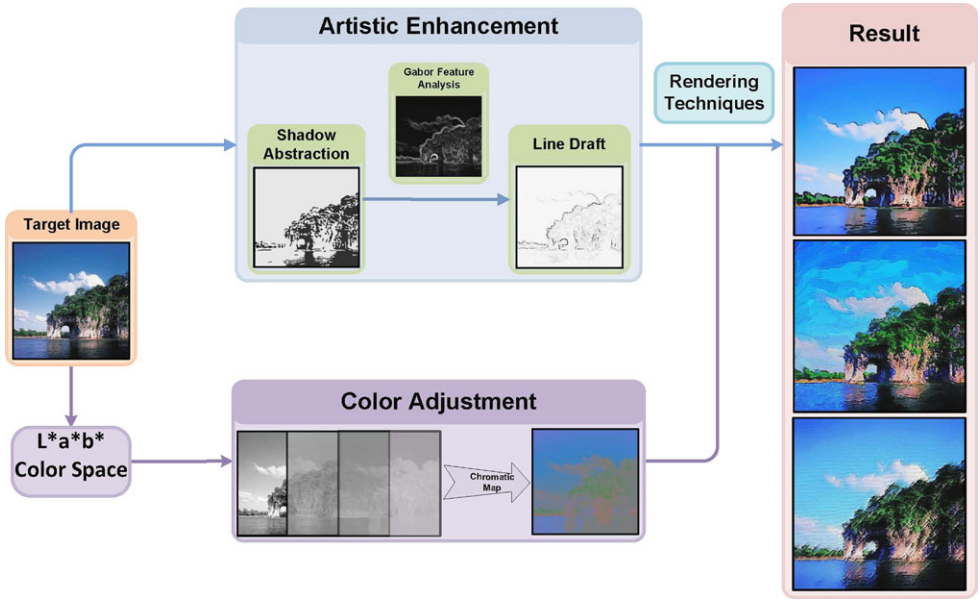


Fig. 2 The artistic preprocessing framework

[37] and Zhang et al. [44] noticed the difference of color styles between photos and real paintings, they only considered the brightness of the target image [37] or required an extra artistic template [44].

Our work is also related to some line emphasis research. Gooch et al. [7] employed the DoG edge detector and a brightness perception model to generate human facial illustrations. Raskar et al. [30] designed a novel camera with multiple flashes which can distinguish the depth edges from intensity edges, and based on the detected edges, the system can generate stylized and animated images. Kang et al. [17] proposed a video-flow based DoG edge model, and focused on generating consecutive and coherent line drawings. These edge-detection based techniques mainly rely on the local color contrast of the image, and as a result they do not distinguish between texture lines in regions with high color contrast and structural contours that are particularly important from the artistic viewpoint.

3 Algorithm

Research in human visual systems [31, 32] indicate that the significant contours are usually first observed when viewing an object. Then, the object's global appearance is perceived by its overall color distribution, after which the fine details and textures are reflected by the local light–shadow contrast. The real painting process is similar: a line draft is often first created to constrain the overall composition of a picture and to enhance the significant contours and details at different levels; then, vivid colors with emotional contrast are painted to express intense emotions; and finally, the shadow areas are refined to present either realistic or abstract results. Our artistic preprocessing technique mainly focuses on artistic enhancement and color adjustment, as shown in Fig. 2. As the enhancement of the light–shadow contrast will change image brightness, we perform this operation in the first step and then compensate the changes in the color adjustment scheme.

3.1 Artistic enhancement

The artistic enhancement mainly treats light–shadow contrast, significant contour and shadow line emphasis, which are detailed in the following subsections.

3.1.1 Shadow abstraction

The shadow abstraction is implemented via two steps: (1) each pixel is assigned a flag to denote whether it belongs to a light area or a shadow area; (2) the flagged pixels are clustered and the resulting shadow regions are further simplified to form the abstract shadow layer. The motivation of our shadow abstraction algorithm partially comes from artistic binarization such as in Mould and Grant [27], Xu and Kaplan [41], but they did not provide an effective simplification strategy.

Extraction A simple way to label pixels with light or shadow flags is to set up a threshold, either automatically (see e.g. [23, 29, 33]) or interactively. It is desired that the labeling process refers not only to the global average lightness but also to the local contrast and the spatial coherence. Also, in order to satisfy different preferences, it is necessary to allow the user to specify the overall light or dark tone.

Therefore, an energy function combining the above considerations is designed and we employ graph cut [1] to find the optimal labeling of light and shadow.

Let $I(p)$ denote the intensity of a pixel p and L_p denote its shadow status, i.e. $L_p = 0$ indicating shadow and $L_p = 1$ indicating light. The cost function of each pixel p is formulated as

$$C_p(L_p, L_q) = D_p(L_p) + \alpha_s \cdot S_{(p,q)}(L_p, L_q) \tag{1}$$

where $\alpha_s \in [0, 1]$ is a balancing factor, and the data term $D_p(L_p)$ and the smooth term $S_{(p,q)}(L_p, L_q)$ are defined as

$$D_p(L_p) = \big(\alpha_g \cdot \big(I(p) - \phi \cdot M_g(I)\big) + (1 - \alpha_g) \cdot \big(I(p) - M_l(N)\big)\big) \cdot \mathrm{sign}_D(L_p) \tag{2}$$

and

$$S_{(p,q)}(L_p, L_q) = \sum_{q \in N(p)} \mathrm{sign}_E(L_p, L_q) \cdot \big|I(p) - I(q)\big|. \tag{3}$$

The data term $D_p(L_p)$ (2) indicates the light–shadow status of each pixel and the smooth term $S_{(p,q)}(L_p, L_q)$ (3) indicates the influence from neighboring pixels. In Eq. (2), α_g is a trade-off factor which determines how the global average intensity $M_g(I)$ affects the results, $M_l(N)$ is the local average intensity of the neighborhood centering at pixel p (the size of the neighborhood is set as 5×5 in our experiments), ϕ is an adjustable parameter that allows the user to determine the black or white tone of the shadow layer, and the sign function $\mathrm{sign}_D(L_p)$ in the $D_p(L_p)$ term equals 1 if $L_p = 0$ and equals -1 if $L_p = 1$. In the smooth term $S_{(p,q)}(L_p, L_q)$, $N(p)$ represents the neighborhood of pixel p, and the sign function $\mathrm{sign}_E(L_p, L_q)$ equals 0 if $L_p = L_q$ and equals 1 otherwise. Following the saliency detection technique in Cheng et al. [2], the factor α_g can be determined by the saliency $s(p)$ of the pixel p as $1 - s(p)/255$.

Finally, the classic graph cut is employed to find an optimal labeling scheme which minimizes the total cost of all the pixels, and construct a binary map that approximately depicts the shadow distribution.

Simplification The binary image obtained from the above extraction step cannot be used as the shadow map because of the mottled artifacts caused by the noise. In order to remove these artifacts and form a smooth shadow map, the mean-shift clustering algorithm [4] is employed to obtain an initial shadow region, which is then further simplified to simulate the rectilinear shadow line often appearing in real artworks.

For each separate shadow region obtained from the mean-shift clustering algorithm, the contour pixels are extracted and ordered to form a list $\ell(R)$, and the curvature $c(p)$ at each point p in the list is then calculated. Our goal is to find

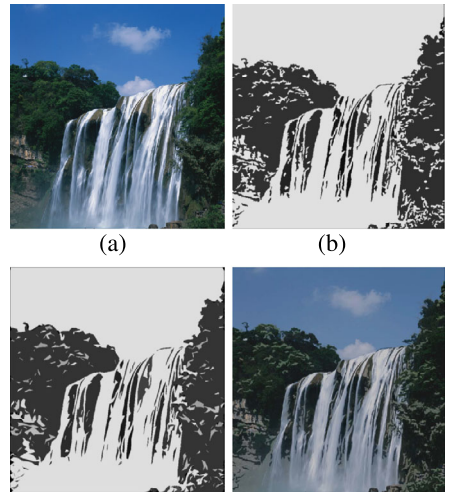


Fig. 3 Shadow abstraction: (**a**) the input image; (**b**), (**c**) the extracted shadow layers at different simplification levels; (**d**) the result after overlaying the shadow layer

a down-sampled list $\hat{\ell}(R)$ such that

$$\arg\min_{\hat{\ell}(R)} \Big(\sum_{p \in \hat{\ell}(R)} c(p) - \sum_{p \in \ell(R)} c(p) \Big)^2. \tag{4}$$

The down-sampling factor for the list $\hat{\ell}(R)$ is set as 2, and this coarsening process can be performed repeatedly to obtain the shadows at different simplification levels (denoted as L_s). After obtaining the simplified contours, each separate shadow region is filled with the mean value of the pixels belonging to this region. Figure 3 shows an example of shadow abstraction, where Fig. 3(a) is the input photo, Fig. 3(b) and (c) the extracted shadow layers at different simplification levels, and Fig. 3(d) the result after adding the shadow map to the original image.

To examine the performance of shadow abstraction, we compare it with the artistic binarization technique [27] in Fig. 4, where the left column shows the input images, the middle column the results produced by Mould and Grant [27], and the right column our results. Unlike the artistic binarization method which focuses on finding the optimal black-and-white version of the inputs, our shadow abstraction technique generates an abstract and simplified shadow layer with sharp contrast between light and dark, which is more suitable for artistic enhancement.

3.1.2 *Line emphasis*

Differently from previous line emphasis techniques [5, 12, 18, 36, 38, 39, 43] that are based on edge detectors, we present a new approach based on Gabor feature analysis, which provides multi-level line maps allowing regions with different visual significance to be emphasized at different levels.

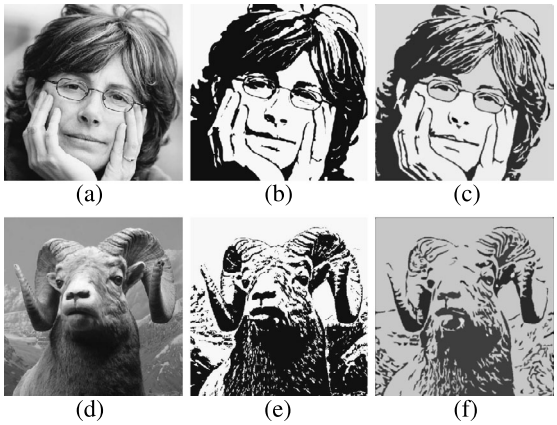


Fig. 4 Comparison of shadow abstraction and artistic binarization. The *left column* shows the input images, the *middle column* the results of previous binarization technique, and the *right column* our shadow abstraction results

The Gabor filter was first introduced by Dennis Gabor, and for 2D image signals the Gabor-energy can be formulated as [34]:

$$e_{\lambda,\theta,\sigma}(x,y) = \sqrt{R_{\lambda,\theta,\sigma,0}(x,y)^2 + R_{\lambda,\theta,\sigma,\frac{\pi}{2}}(x,y)^2}, \tag{5}$$

where $R_{\lambda,\theta,\sigma,0}(x,y)$ and $R_{\lambda,\theta,\sigma,\frac{\pi}{2}}(x,y)$ are the real and imaginary Gabor filter responses. The superposition response e_λ is computed as the pixel-wise maximum response in all orientations (i.e. $\theta = 0, \theta = \frac{\pi}{8}, \ldots, \theta = \frac{7\pi}{8}$), where σ is set to 1.0 in all calculations.

Our main idea is to identify the significant contours and the detail regions, and then emphasize them at different levels using lines of various sizes. With the change of spatial frequencies λ, the Gabor responses e_λ vary differently for the texture-like and the contour-like areas. Specifically, for texture areas the response decreases as the scale increases, while for contours the response always increases or stays unchanged. An example is given in Fig. 5(a–c), where the superposition Gabor responses of Fig. 3(a) are computed at three different spatial frequencies, $\lambda = \sqrt{2}, 4\sqrt{2}, 7\sqrt{2}$.

Based on this observation, we compute the superposition Gabor response at seven different frequencies, denoted by $e_{\lambda_1}, e_{\lambda_2}, \ldots, e_{\lambda_7}$, and use them as the main technical references to extract the line drafts of input images. Specifically, the line draft of the detail regions L_D is extracted as

$$L_D(x,y) = \sum_{i=1}^{6} E_i(x,y)\mu\big(E_i(x,y)\big), \tag{6}$$

where $E_i(x,y)$ is the initial estimation of the line draft at the scale λ_i, and $\mu(E_i(x,y))$ is the adjustment term due to spatial correlation. The line drafts at different scales are calculated by comparing the difference between superposition Gabor-energy responses at neighboring scales, i.e.

$$E_i(x,y) = \big(e_{\lambda_i}(x,y) - e_{\lambda_{i+1}}(x,y)\big)$$

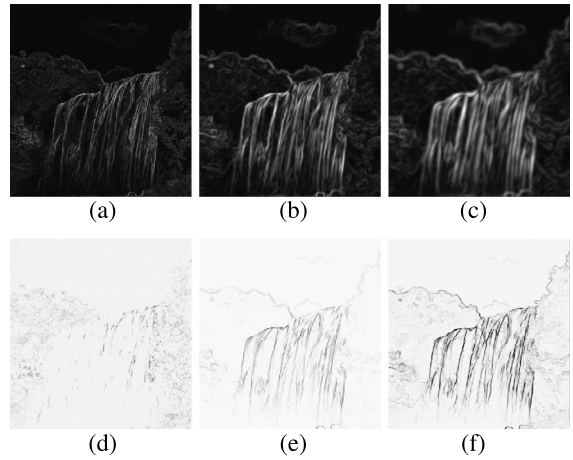


Fig. 5 Line emphasis illustration: (**a**), (**b**), (**c**) superposition Gabor responses of Fig. 3(a) at spatial frequencies $\lambda = \sqrt{2}$, $\lambda = 4\sqrt{2}$ and $\lambda = 7\sqrt{2}$, respectively; (**d**), (**e**) the separation of the detail regions and the significant contours; (**f**) the line draft formed for the input image

$$\times\ \mathrm{sign}\big(e_{\lambda_i}(x,y) - e_{\lambda_{i+1}}(x,y)\big), \tag{7}$$

where $\mathrm{sign}(z)$ denotes the sign function that equals 1 if $z > 0$ and equals 0 if $z \le 0$. The response of each pixel is influenced by its neighbors through the adjustment term

$$\mu\big(E_i(x,y)\big) = \sum\nolimits_{(x',y')\in\Omega} E_i\big(x',y'\big), \tag{8}$$

where Ω denotes the neighborhood of the pixel $p(x,y)$. The radius of Ω is set as 2 in our implementation.

Then the line draft of the contour regions can be obtained as

$$L_C = e_{\lambda_1} - \gamma \cdot G_\sigma * L_D, \tag{9}$$

where e_{λ_1} is the superposition Gabor response computed for the lowest spatial frequency $\lambda_1 = \sqrt{2}$ containing all texture information with fine details, $G_\sigma * L_D$ is the convolution of a Gaussian kernel G_σ and the image details L_D obtained in Eq. (6), and $\gamma \in [1.0, 8.0]$ is an adjustable parameter controlling the sensitivity of the contour regions (with 1 indicating the highest sensitivity of contour detection and 8 the lowest sensitivity).

Finally, the line draft of the image can be obtained by blending the image details L_D, the contours L_C, and the details coming from the shadow map L_{D_s} (which is computed in a similar way as Eq. (6)). Figure 5(d) shows the detail features (i.e. L_D) extracted from Fig. 3(a), Fig. 5(e) shows the significant contours (i.e. L_C), and Fig. 5(f) shows the blended line draft for the input image.

Previous edge detection works such as Gooch et al. [7], Winnemöller et al. [38] and Kang et al. [17] aim to produce uniform line maps, and can generate impressive results, while our approach has different goal that tries to separate the details and the significant contours, so regions

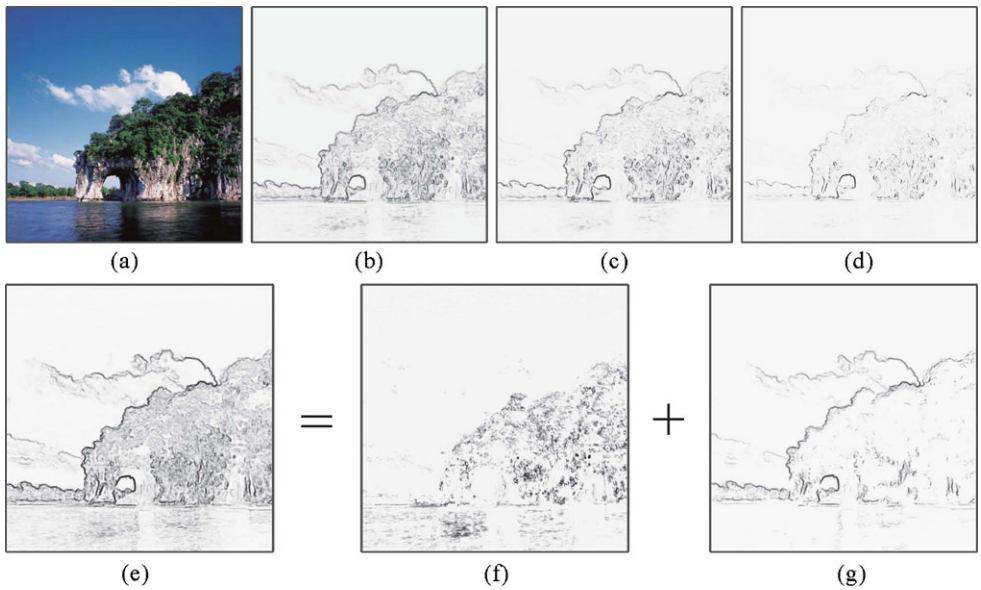


Fig. 6 Comparison with DoG edge detector: (**a**) the input; (**b**)–(**d**) DoG edge detection results with different sensitivity ($\tau = 0.900, 0.980.0.998$); (**e**) the superposition Gabor responses e_{λ_1} with frequency $\lambda_1 = \sqrt{2}$; (**f**), (**g**) separated detail and contour part L_D and L_C with $\gamma = 6.0$

with different visual significance can be emphasized at different levels flexibly. Figure 6 shows a comparison with the DoG (difference-of-Gaussians) edge detector (which is widely applied in edge detection such as Gooch et al. [7], Winnemöller et al. [38] and Kang et al. [17]) with different degrees of detection sensitivity. From the results it can be viewed that, for DoG edge detector in Fig. 6(b–d), details can be gradually removed as the decrease of the detection sensitivity; however, significant contours also become less prominent. While our approach can separate details from the initial superposition Gabor response and preserve coherent and clear contours, as shown in Fig. 6(e–g).

3.2 Color adjustment

Real artworks, which appear intense and emotional, often feature vivid colors with strong contrast. To achieve such effect, a color adjustment scheme is proposed to help users create artistic color style. Differently from simple color-saturation adjustment, the new scheme emphasizes the "feature" of each color to produce an emotional contrast. Together with a compensation technique that is consistent with the real color toning process, it provides a closer imitation of real artworks' color features.

First, a chromatic map is extracted by removing the influence of lightness. Specifically, after decomposing the input image in the $L^*a^*b^*$ color space, its chromatic map can be obtained by merging the original a^* and b^* channels with a neutral light channel set as the mid-value of the whole range. Using this map, the R, G, B color components of each pixel are enhanced as

$$C'(p) = C_I(p) \cdot \frac{1 + \tanh(\rho \cdot (C_M(p) - 128))}{2} \tag{10}$$

where $C_I(p)$ is the original color of the result obtained in Sect. 3.1.1 at pixel p, $C_M(p)$ is the color of the corresponding pixel in the chromatic map, and ρ is an adjustable factor controlling the enhancement level (typical values are in the range of [0.005, 0.2]).

The artistic enhancement and color adjustment may cause undesired changes to the overall brightness of the input image. For real paintings, artists commonly adjust the overall tone of the canvas by adding white or black color. Inspired by this, a tone adjustment scheme is designed and we operate it separately in each color channel of the RGB color space:

$$\hat{C}(p) = C'(p) + 2^{\kappa} \cdot C_w \cdot \left(1 - \left|C'(p) - \mu_c\right| / \mu_c\right), \tag{11}$$

where $\hat{C}(p)$ is the adjusted color of each pixel p, $C'(p)$ the current color before adjustment, $\kappa \in [1.0, 4.0]$ a tuning parameter which controls the compensation extent, C_w the white color (i.e. 255 for each color channel), μ_c the average value in each color channel. The adjusted color $\hat{C}(p)$ is reset to the boundary value (i.e. 0 or 255) if its calculated value falls outside the range of [0,255]. Using the above color adjustment scheme, a stronger tone adjustment is delivered to pixels that have an average color intensity.

Figure 7 shows the comparison of our color adjustment scheme and the saturation correction operation in Photoshop CS4. Figure 7(a) is the input image, Fig. 7(b) and (c) are the Photoshop results of color saturation correction processed at different degrees, and Fig. 7(d) is our result. Differently from the simple linear saturation correction in Photoshop CS4, our adjustment scheme concentrates more on emphasizing the "feature" of each color. That is, the advantage component of a color is amplified and other components are suppressed, so that the overall color saturation can keep balanced while the overall color style appears lively with sharp

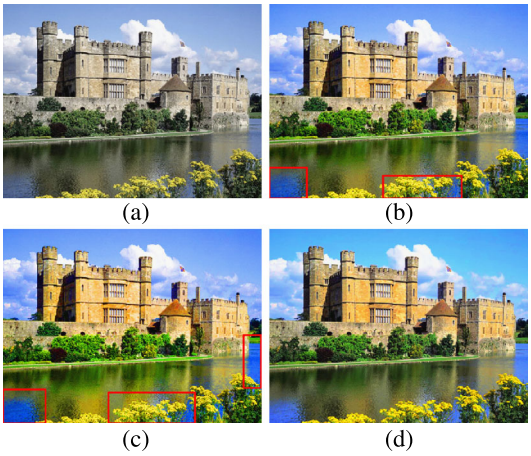


Fig. 7 Comparison of our color adjustment scheme and linear saturation correction: (**a**) the input image; (**b**), (**c**) linear saturation correction by Photoshop CS4; (**d**) our adjustment result

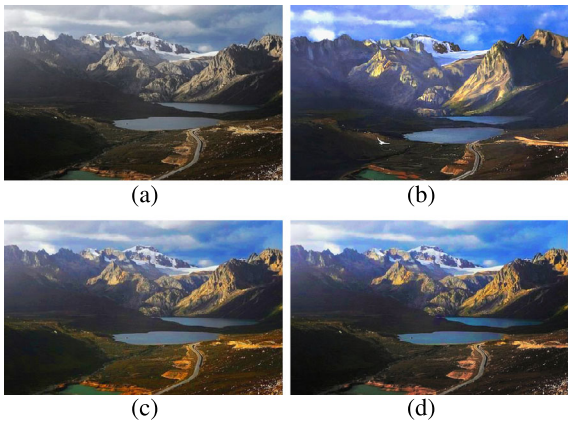


Fig. 8 Comparison of a recent color transfer method and our color adjustment scheme: (**a**) the input image; (**b**) an artistic version created by a professional artist; (**c**) the result produced by Xiao and Ma [40] using (**b**) as the template along with some manual adjustment (total processing time: 33 seconds); (**d**) the result produced by our color adjustment scheme (total adjustment time: 9 seconds)

contrast. The new technique is more consistent with the real toning process and the artifact of oversaturation (as shown in the red boxes of Fig. 7(b) and (c)) can be avoided.

Figure 8 shows another comparison with the recent color transfer method proposed by Xiao and Ma [40]. Figure 8(a) shows the input image, and Fig. 8(b) shows an artistic version created by a professional artist, which is taken in this comparison as the ground truth for artistic enhancement. Figure 8(c) is the result from Xiao and Ma [40], where Fig. 8(b) is used as the template to transfer its color style to the input image and some manual color adjustments are applied using Photoshop CS4 to achieve the best imitation of Fig. 8(b). Figure 8(d) is our result, which is obtained directly from the input image through the proposed automatic color adjustment scheme. It can be seen that our method provides a color style with sharp contrast which is more consistent with the professional artist's result, such as the color of the lake and the contrast between land and sky. In addition, the proposed color adjustment scheme is much more efficient and significantly reduces the processing time from 33 to 9 seconds.

4 Experimental results

In this section, several experiments and comparisons are presented to examine the performance of the new preprocessing technique. All experiments are performed on a PC platform with an Intel 3.0 GHz Dual-Core CPU and 4 GB RAM. For an image with an approximate size of 600×800, the average running time is 4 seconds.

Artistic preprocessing We first examine how the proposed preprocessing technique can benefit existing NPR systems. As shown in Fig. 9, four input images of dramatically different scenes (portrait, landscape, seascape, river land) are chosen for the demonstration. In Fig. 9, the left column shows the input pictures, the middle column shows the results directly obtained from existing NPR systems, and the right column shows the results from existing NPR systems after applying the proposed artistic preprocessing technique. From the top to the bottom row, we compare respectively the rendering software Toon-FX (the free trial version used for test purpose, so the watermark remained) released by Winnemöller et al. [38], the stroke-based rendering platform proposed by Huang et al. [13, 14], the poster edge filter and the crayon filter in Photoshop CS4. The comparison shows that the proposed artistic preprocessing technique is versatile and compatible with existing stylization and stroke-based rendering systems, and can help produce better rendering results without additional input or user intervention. Our artistic preprocessing technique can also independently produce stylization results with different artistic effects, and more results can be viewed in Fig. 10.

User evaluation Quantitative measure of aesthetics remains a challenging task [10], although for specific image applications some recent progress has been made in this direction [15, 25, 26]. Therefore, for a full evaluation of the new artistic preprocessing technique, we design a user study involving both armature users with some art knowledge and professional art students. Ten groups of rendering results are produced, and each group contains two images, one directly rendered with some existing stylization and painterly rendering systems [13, 19, 38], and the other generated by combining with our new preprocessing step. First, these results were presented to 20 voluntary participants (10 males and 10 females) who have some art knowledge. The users were

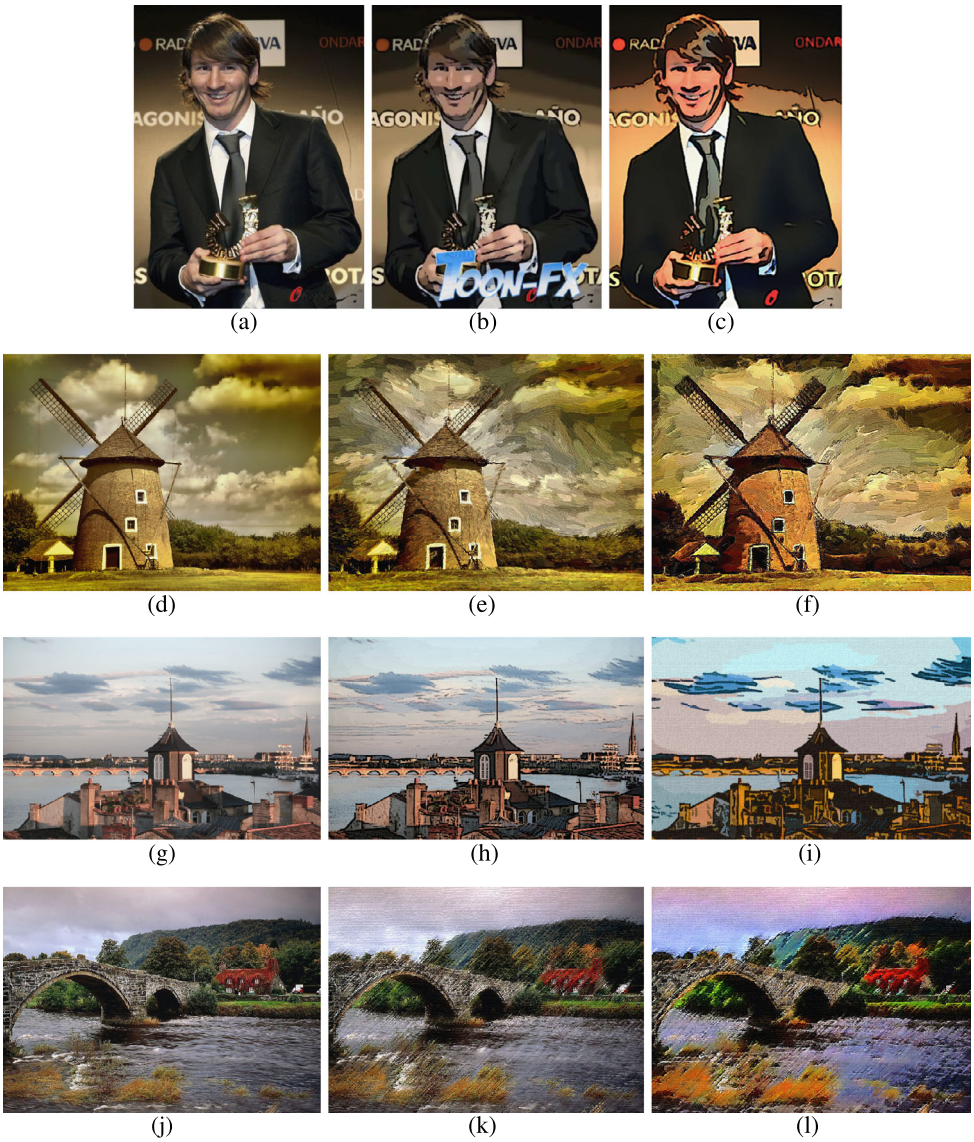


Fig. 9 Artistic preprocessing benefits existing stylization and painterly rendering systems. The *first column*, including (**a**), (**d**), (**g**) and (**j**) shows the input images. The *second column* shows the rendering results directly obtained from some existing non-photorealistic rendering systems, including (**b**), the Toon-Fx software developed by Winnemöller et al. [38]; (**e**) the stroke-based painterly rendering platform by Huang et al. [13, 14]; (**h**) the poster edge filter in Photoshop CS4 and (**k**) the crayon filter in Photoshop CS4. The *right column*, including (**c**), (**f**), (**i**) and (**l**) shows the results from existing non-photorealistic rendering systems after applying the proposed artistic preprocessing technique

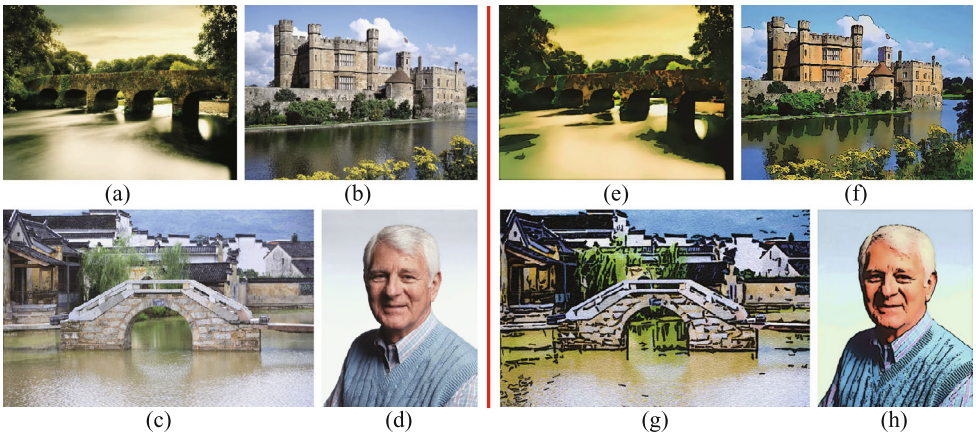


Fig. 10 Stylization results using the proposed artistic preprocessing technique. The *left part* including (**a**–**d**) shows the input images, and the *right part* including (**e**–**h**) shows the results of our approach

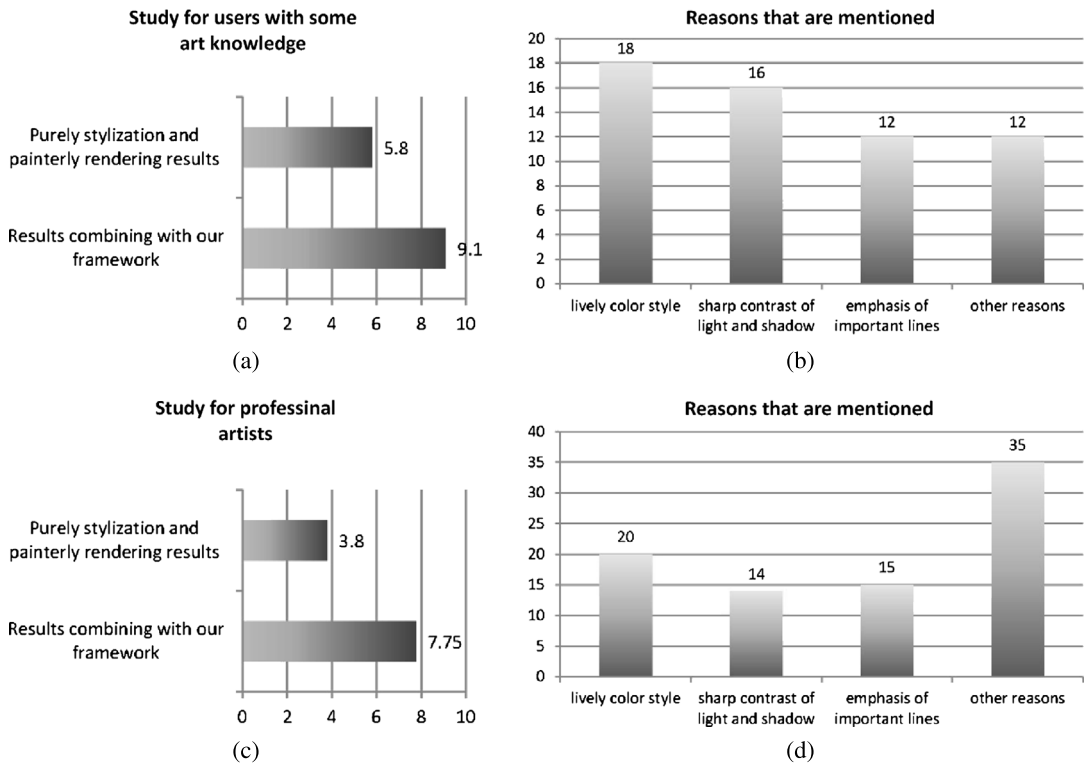


Fig. 11 The user study results

asked to measure the artistic effect of each result and give a score out of 10 (with 1 indicating the worst and 10 the best). The user evaluation results are shown in Fig. 11(a, b). It can be seen in Fig. 11(a) that users generally agree that the new preprocessing step does improve considerably the performance of existing painterly rendering methods and produces better artistic effects. From Fig. 11(b) it is seen that almost all users consider that the overall color style is the most important reason for the visual effect improvement, and more than half of them also point out the reason of abstracted shadow layer and sharp contrast along with the emphasized lines. The same comparison results were also presented to 20 art students (10 males and 10 females) for more professional feedbacks. The evaluation results are shown in Fig. 11(c, d). It can be seen from Fig. 11(c) that the art students are generally more critical for the rendering results, giving somewhat lower scores. As shown in Fig. 11(d), all artist users note the improvement of color style, and over 70 % of them point out the reason of shadow lines. It is also noticed that the art students address more aspects beyond the current work, which may imply a significant scope of further exploration for artistic enhancement.

Parameter discussion The visual effect of the proposed artistic preprocessing technique is primarily controlled by six parameters, which can be divided into 3 groups, as shown in Table 1. The parameters α_s, ϕ and L_s control shadow abstraction. Specifically, $\alpha_s \in [0, 1]$ controls the coherent degree of shadow layer with a lager value producing a more coherent result, $\phi \in [0, 1]$ determines the overall black or white tone of the shadow layer with a lager value producing a brighter effect, and $L_s \in [0, 8]$ determines the simplification degree of shadow layer with a lager value producing a simpler result. The parameter $\gamma \in [1, 8]$ controls the sensitivity of contour detection and a larger value means lower sensitivity. The parameters $\rho \in [0.008, 0.2]$ and $\kappa \in [1, 4]$ control the color adjustment result, and larger values of ρ and κ provide more colorful and brighter results. Table 1 lists the parameter setting for the results shown in Figs. 9 and 10, where "N/A" means the corresponding step is not applied.

Table 1 Control parameters for Figs. 9 and 10

	α_s	ϕ	L_s	γ	ρ	κ
Fig. 9(c)	0.20	0.70	3	3.30	0.012	1.9
Fig. 9(f)	0.27	0.64	2	6.25	0.008	1.5
Fig. 9(i)	0.13	0.77	3	N/A	0.006	1.7
Fig. 9(l)	0.34	0.63	3	N/A	0.011	1.6
Fig. 10(e)	0.23	0.67	3	1.50	0.011	1.8
Fig. 10(f)	0.25	0.57	2	1.92	0.011	1.6
Fig. 10(g)	0.16	0.68	4	2.74	0.006	1.6
Fig. 10(h)	0.41	0.57	2	1.77	0.009	1.9

5 Conclusions

An artistic preprocessing technique for image stylization and painterly rendering applications is proposed to enhance the artistic impression of the rendering result. The practical method is consistent with the procedure of hand drawing, and through artistic enhancement and color adjustment it allows much stronger artistic effects to be achieved using standard rendering methods. Demonstrated by various examples, the artistic preprocessing operation can significantly improve the light–shadow contrast, the line emphasis, and the color scheme. In addition, the proposed preprocessing technique can also be used as an independent tool to generate different types of stylization results.

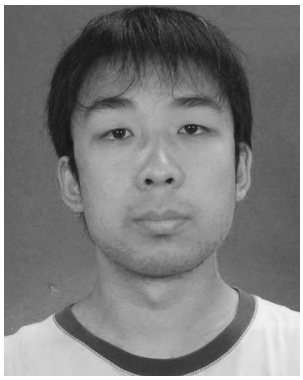
Extending the proposed method into videos may be an interesting direction for future work, for which the temporal coherence of the shadow abstraction and line draft generation between frames can be a new challenge. Another interesting direction is to explore other artistic styles such as shape exaggeration.

Acknowledgements This work is supported by Tsinghua—Tencent Joint Laboratory for Internet Innovation Technology under Grant No. 2012-01 and the National Natural Science Foundation of China (No. 60970068). The authors would also like to thank the support from the International Joint Project from the Royal Society of UK (No. JP100987.)

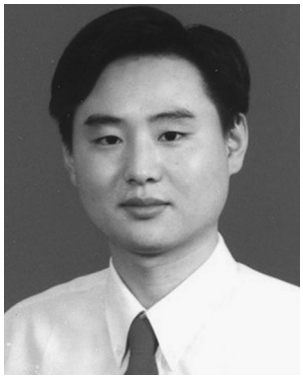
References

1. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. IEEE Trans. Pattern Anal. Mach. Intell. **23**(11), 1222–1239 (2001)
2. Cheng, M., Zhang, G., Mitra, N., Huang, X., Hu, S.: Global contrast based salient region detection. In: Proceedings of Computer Vision and Pattern Recognition. IEEE Press, New York (2011)
3. Collomosse, J., Rowntree, D., Hall, P.: Stroke surfaces: temporally coherent artistic animations from video. IEEE Trans. Vis. Comput. Graph. **11**(4), 540–549 (2005)
4. Comaniciu, D., Meer, P.: Mean shift: a robust approach toward feature space analysis. IEEE Trans. Pattern Anal. Mach. Intell. **24**(5), 603–619 (2002)
5. Cong, L., Tong, R., Dong, J.: Selective image abstraction. Vis. Comput. **27**(3), 187–198 (2011)
6. DeCarlo, D., Santella, A.: Stylization and abstraction of photographs. ACM Trans. Graph. **21**(3), 769–776 (2002)
7. Gooch, B., Reinhard, E., Gooch, A.: Human facial illustrations: creation and psychophysical evaluation. ACM Trans. Graph. **23**(1), 27–44 (2004)
8. Hays, J., Essa, I.: Image and video based painterly animation. In: Proceedings of the International Symposium on Non-Photorealistic Animation and Rendering, pp. 113–120. ACM, New York (2004)
9. Hertzmann, A.: Painterly rendering with curved brush strokes of multiple sizes. In: Proceedings of SIGGRAPH, pp. 453–460. ACM, New York (1998)
10. Hertzmann, A.: Non-photorealistic rendering and the science of art. In: Proceedings of the 8th International Symposium on Non-Photorealistic Animation and Rendering, pp. 147–157 (2010)
11. Hu, S.-M., Chen, T., Xu, K., Cheng, M.-M., Martin, R.R.: Internet visual media processing: a survey with graphics and vision applications. Vis. Comput. **29**(5), 393–405 (2013)
12. Huang, Y., Ralph, M., Paul, L., Meng, X., Yang, C.: Expressive line drawings of human faces from range images. Sci. China, Ser. F, Inf. Sci. **52**(2), 295–307 (2009)
13. Huang, H., Zhang, L., Fu, T.: Video painting via motion layer manipulation. Comput. Graph. Forum **29**(7), 2055–2064 (2010)
14. Huang, H., Fu, T., Li, C.: Painterly rendering with content-dependent natural paint strokes. Vis. Comput. **27**(9), 1–11 (2011)
15. Hulusic, V., Debattista, K., Chalmers, A.: Smoothness perception. Vis. Comput. **29**(11), 1159–1172 (2013)
16. Kagaya, M., Brendel, W., Deng, Q., Kesterson, T., Todorovic, S., Neill, P., Zhang, E.: Video painting with space-time-varying style parameters. IEEE Trans. Vis. Comput. Graph. **17**, 74–87 (2010)
17. Kang, H., Lee, S., Chui, C.: Coherent line drawing. In: Proceedings of the International Symposium on Non-Photorealistic Animation and Rendering, pp. 43–50. ACM, New York (2007)
18. Kang, H., Lee, S., Chui, C.: Flow-based image abstraction. IEEE Trans. Vis. Comput. Graph. **15**(1), 62–76 (2009)
19. Kyprianidis, J., Kang, H.: Image and video abstraction by coherence-enhancing filtering. Comput. Graph. Forum **30**(2), 593–602 (2011)
20. Kyprianidis, J.E., Collomosse, J., Wang, T., Isenberg, T.: State of the art: a taxonomy of artistic stylization techniques for images and video. IEEE Trans. Vis. Comput. Graph. **19**(5), 866–885 (2013)
21. Lee, H., Lee, C.H., Yoon, K.: Motion based painterly rendering. Comput. Graph. Forum **28**(4), 1207–1215 (2009)
22. Li, P., Sun, H.Q., Sheng, B., Shen, J.B.: Image stylization with enhanced structure on GPU. Sci. China, Inf. Sci. **55**(5), 1093–1105 (2012)
23. Liao, P., Chen, T., Chung, P.: A fast algorithm for multilevel thresholding. J. Inf. Sci. Eng. **17**(5), 713–728 (2001)
24. Litwinowicz, P.: Processing images and video for an impressionist effect. In: Proceedings of SIGGRAPH, pp. 407–414 (1997)
25. Liu, L., Chen, R., Wolf, L., Cohen-Or, D.: Optimizing photo composition. Comput. Graph. Forum **29**(2), 469–478 (2010)
26. Liu, Y., Luo, X., Huan, Y., Chen, W., Fu, X.: Image retargeting quality assessment. Comput. Graph. Forum **30**(2), 583–592 (2011)
27. Mould, D., Grant, K.: Stylized black and white images from photographs. In: Proceedings of the International Symposium on Non-Photorealistic Animation and Rendering, pp. 49–58. ACM, New York (2008)
28. O'Donovan, P., Hertzmann, A.: Anipaint: interactive painterly animation from video. IEEE Trans. Vis. Comput. Graph. **18**(3), 475–487 (2012)
29. Otsu, N.: A threshold selection method from gray-level histograms. Automatica **11**(3), 285–296 (1975)
30. Raskar, R., Tan, K., Feris, R., Yu, J., Turk, M.: Non-photorealistic camera: depth edge detection and stylized rendering using multi-flash imaging. ACM Trans. Graph. **23**(3), 679–688 (2004)
31. Sun, Y., Fisher, R.: Object-based visual attention for computer vision. Artif. Intell. **146**(1), 77–123 (2003)
32. Torralba, A., Oliva, A., Castelhano, M., Henderson, J.: Contextual guidance of eye movements and attention in real-world scenes: the role of global features in object search. Psychol. Rev. **113**(4), 766 (2006)
33. Tsai, W.: Moment-preserving thresholding: a new approach. Comput. Vis. Graph. Image Process. **29**(3), 377–393 (1985)
34. Turner, M.R.: Texture discrimination by Gabor functions. Biol. Cybern. **55**(2), 71–82 (1986)
35. Wang, J., Xu, Y., Shum, H., Cohen, M.: Video tooning. In: Proceedings of ACM SIGGRAPH, pp. 574–583. ACM, New York (2004)

36. Wang, S., Cai, K., Lu, J., Liu, X., Wu, E.: Real-time coherent stylization for augmented reality. Vis. Comput. **26**(6), 445–455 (2010)
37. Wen, F., Luan, Q., Liang, L., Xu, Y., Shum, H.: Color sketch generation. In: Proceedings of the International Symposium on Non-Photorealistic Animation and Rendering, pp. 47–54. ACM, New York (2006)
38. Winnemöller, H., Olsen, S.C., Gooch, B.: Real-time video abstraction. ACM Trans. Graph. **25**(3), 1221–1226 (2006)
39. Wong, F.J., Takahashi, S.: Abstracting images into continuous-line artistic styles. Vis. Comput. **29**(6–8), 729–738 (2013)
40. Xiao, X., Ma, L.: Gradient-preserving color transfer. Comput. Graph. Forum **28**(7), 1879–1886 (2009)
41. Xu, J., Kaplan, C.: Artistic thresholding. In: Proceedings of the International Symposium on Non-Photorealistic Animation and Rendering, pp. 39–47. ACM, New York (2008)
42. Zeng, K., Zhao, M., Xiong, C., Zhu, S.: From image parsing to painterly rendering. ACM Trans. Graph. **29**(1), 2 (2009)
43. Zhang, S., Chen, T., Zhang, Y., Hu, S., Ralph, M.: Video-based running water animation in Chinese painting style. Sci. China, Ser. F, Inf. Sci. **52**(2), 162–171 (2009)
44. Zhang, S.H., Li, X.Y., Hu, S.M., Martin, R.R.: Online video stream abstraction and stylization. IEEE Trans. Multimed. **13**(6), 1286–1294 (2011)
45. Zhao, M., Zhu, S.: Sisley the abstract painter. In: Proceedings of the 8th International Symposium on Non-Photorealistic Animation and Rendering, pp. 99–107. ACM, New York (2010)



Yu Zang is currently a doctoral student of School of Electronics and Information Engineering, Xian Jiaotong University. He received the Bachelors degree in Xian Jiaotong University in 2008. His main research interests include image and video processing.



Hua Huang is a professor in School of Computer Science and Technology, Beijing Institute of Technology. He received his BS and PhD degrees from Xian Jiaotong University in 1996 and 2006, respectively. His main research interests include image and video processing, computer graphics, pattern recognition, and machine learning.

Chen-Feng Li is currently a Lecturer at the College of Engineering of Swansea University, UK. He received his BEng and MSc degrees from Tsinghua University, China in 1999 and 2002 respectively and received his PhD degree from Swansea University in 2006. His research interests include computer graphics and numerical analysis.