Multimed Tools Appl (2018) 77:16113–16130
DOI 10.1007/s11042-017-5190-z

Automatic portrait oil painter: joint domain stylization for portrait images

Saboya Yang[1] · Shuai Yang[1] · Wenhan Yang[1] · Jiaying Liu[1]

Received: 6 February 2017 / Revised: 29 August 2017 / Accepted: 1 September 2017 /
Published online: 11 September 2017

Abstract Everyone has the dream of being in the center of famous art paintings, admired by numerous future generations. However, the dream came true at a huge cost of the painter's commission in old days. In our paper, another practical choice is provided for everyone to achieve that dream – an automatic portrait oil painter transferring some artistic styles from one single reference painting. To address this issue, we propose a joint-domain image stylization approach, particularly for portrait oil paintings. From the view of artistic appreciation, we analyze an amount of oil painting art works and summarize three critical factors to depict the figure, i.e. color, structure and texture. Based on this point, we separate and represent an artistic work into these three domains. Then, considering their intrinsic properties and following an art creation route, we propose the corresponding approaches to jointly model and transfer the features in these domains. First, a swatch-based color adjustment is proposed to recolor the tone of the input image based on semantic regions corresponding to the references. Second, the main structures of the input image is maintained by sparse reconstruction. Third, a coarse-to-fine texture synthesis is used to enhance the detail oil painting patterns. Extensive experimental results demonstrate that the proposed method achieves desirable results compared with state-of-the-art methods in not only transferring the styles from references but also keeping consistent contents with the given portrait.

Keywords Image stylization · Saliency aware · Dictionary learning · Texture synthesis

1 Introduction

Nowadays it is becoming popular to share photos through social media. With the trend that more and more people prefer uploading their photos decorated with special styles generated

✉ Jiaying Liu
liujiaying@pku.edu.cn

[1] Institute of Computer Science and Technology, Peking University, Beijing, China

by apps, such as Facebook and Instagram instead of the original ones, there is a great demand for a new emerging technology – *Image Stylization*. It creates a painting based on the content of a photo uploaded by users with the styles provided by a series of predefined filters or some external references, such as some fantastic art works or photos. This new technology makes a more dramatic impression and inspires new creativity. Besides, it provides useful side information to serve other applications, such as user profiling [49]. From a technical view, image stylization belongs to the field of nonphotorealistic rendering (NPR) [19] and is regarded as a mapping problem across domains, which aims to transform user-provided images from one style to another.

The seminal work of image stylization starts from the research of Hertzmann [15] in 1998. It first utilized strokes to represent image features and incrementally composed virtual strokes to transfer images into artistic styles. On the basis of this stroke-based method, Zhao and Zhu [52] built an abstract painter to simulate brush strokes of oil paintings and reproduced an image in the oil painting style. This method was improved in [54] by considering perceptual ambiguities of both the scene and individual objects. With only hand-crafted predefined elements that are operated globally, these methods fail to transfer the styles containing multiple elements or complex local variations within one painting.

To handle more general cases, the following researches focused on automatically learning the general mapping between two styles. Jia et al. [18] came up with doing mappings in cross-style feature spaces. Wang and Tang [42] decomposed images into patches and learned a joint photo-sketch model by a multi-scale Markov Random Fields (MRF) model. Wang et al. [43] presented a semi-coupled dictionary learning method to reconstruct the sketch under sparsity constraints. But these learning based methods are on account of a paired training set. Thus, it is hard to apply these methods for handling the case of only one reference, called the *unsupervised style transfer*, which is common when taking art works as reference as shown in Fig. 1. In that case, we have an input portrait image as Fig. 1a and a desirable example as Fig. 1b. We expect to generate a synthesized stylized image with the same style as Fig. 1b and the similar content to Fig. 1a, such as the stylized result in Fig. 1c is generated by the proposed method. Therefore, recent works aim to jointly represent styles and build the mappings across domains for the case of the unsupervised style transfer, with only one reference stylish image. Sunkavalli et al. [41] utilized a multi-scale technique to transfer the appearance of one image to another and composited a harmonized

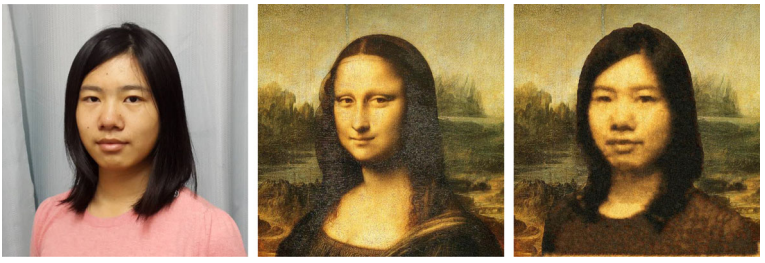


Fig. 1 Illustrations for the concept of *unsupervised style transfer* and the result of the proposed method. **a** Input captured image: a portrait photo captured by cell phone, **b** Input reference image: the famous oil painting Mona Lisa created by Leonardo Da Vinci, **c** Output stylized image: our stylized portrait with the same person as (**a**) and the similar style to (**b**)

image. But this method needed unified masks for harmonization, which are hard to get and lead to the limitation of application scenarios. Lo et al. [26] utilized sparse representation to preserve the original contents but it only transferred simple, repeatable and regular patterns from the reference to the stylized image. There is also a new implementation on neural networks to simulate artistic styles [12]. It obtained representations by convolution networks, which may work inefficiently when the reference does not have obvious streamline textures.

In fact, stacking strokes without damaging the basic structures seems a really representative characteristic of oil paintings. It is challenging to learn the local drawing styles respectively. From the view of artistic appreciation, based on the primal sketch theory of visual perception [28], when looking at a painting, people's eyes take it as a composition of *structure* and *texture*. At the same time, different *color* usage may strongly affect the expression of the painting. This observation motivates us to design an automatic portrait oil painter to handle the task as shown in Fig. 1, via an new proposed automatic style transfer approach for portrait oil painting by jointly modeling structure, texture and color features in their respective domains.

In this paper, we first separately represent an art work in three domains: *structure*, *texture* and *color*. Then, considering the intrinsic properties of the three domains and following an artistic creation route, we propose the corresponding approaches to model and map the features in these domains jointly: 1) color is usually region-consistent, thus a local color transfer method is employed to adjust auto-segmented semantic regions; 2) the main structures of the input image are usually deterministic and contain salient features, such as corners or sharp edges, thus it is maintained by sparse reconstruction; 3) texture is usually locally stochastic but regularly repetitive in global, thus a coarse-to-fine texture synthesis is used separately to bring out more brushwork.

In conclusion, the contributions of this paper lie in following aspects:

- Motivated by the artistic appreciation, artistic elements are summarized into three domains: *color*, *structure* and *texture*, and the portrait oil paintings are presented by the related corresponding features in these three domains.
- Based on this analysis, we propose a joint-domain portrait oil painting stylization approach to transfer the three features across domains jointly, achieving superior performance than state-of-the-art methods.
- For the color domain, we propose an automatic swatch-based color adjustment that segments semantic swatches and transfers colors between these swatches with location information.
- For the structure domain, the fundamental structures acquired from the reference are reconstructed by sparse representation, where the coupled dictionary is trained on the coupled patch set created based on the edge feature correspondences.
- For the texture domain, motivated by the property that textures are locally random but globally consistent, we propose a coarse-to-fine texture synthesis approach. Coarse-grained textures are accumulated by strokes and synthesized on account of intensity similarities and location relations, while fine-grained textures are induced by drawing surfaces and matched by noise pyramids.

The rest of this paper is organized as follows. Related technologies are reviewed in Section 2. The proposed multi-scale portrait oil painting stylization approach is elaborated in Section 3. Experimental results are presented in Section 4. Concluding remarks are given in Section 5.

2 Related works

Image stylization was first proposed by Winkenbach and Salesin [47] targeting to transfer an image into a specified style. The style could be decomposed into different attributes such as color, stroke, contrast, shade, surface and composition. The way to define the attribute features and the methods to decomposition them are the most challenging parts of this task. Feature learning from one resource or multiple resources, and multiple attribute feature fusion also plays an important role in many different application scenarios [24, 27]. Different attributes may lead to different styles such as oil painting [44, 52], watercolor [4, 5], sketch [2, 42], photo portrait [40, 50, 53], cartoon [45] and pen-and-ink [25, 38, 47]. Considering only a subset of these attributes, image stylization could be simplified into some related subproblems, such as color transfer and texture synthesis.

Many recent works focus on the color transfer subproblem. It tends to transform an image to a certain color style [37], which is an esthetically interesting and mathematically difficult problem. Greenfield and House [13] utilized palette color associations to achieve a fast image re-coloring method. Levin [21] asked for user interactions to colorize the gray image. This method was later improved by Huang and Chen [17] on account of landmark pixels. But these methods could not be applied to transfer the reference's colors to the input without user interactions because the local correspondences between images are paid little attention to. Reinhard et al. [34] proposed a robust color transfer method based on the $l\alpha\beta$ color space. However, when the input and the reference contain inhomogeneous color distributions, Reinhard's method may produce unnatural results. Swatches specified by users are then introduced to classify colors [34, 46]. In our method, we propose to acquire swatches automatically by considering the position information. Then by clustering on swatches, semantic regions are segmented to match colors respectively in the $l\alpha\beta$ color space.

Texture synthesis aims to generate complete and high-quality textures from a small set of sample textures [8]. There are mainly two categories of texture synthesis methods: MRF based methods [30] and patch based methods [9]. We focus on patch based methods, which synthesize textures patch by patch and are briefly reviewed here. Hertzmann et al. [16] proposed image analogies to combine the global optimal texture and the local coherent texture together. Ashikhmin extended the search space of the coherent synthesis and provided a fast texture transfer algorithm [1]. It was later enhanced by Lee et al. [20] considering image gradients to append textures. However, these methods mainly reproduced textures with local similarities and neglected the global expressions of artistic styles. Compared with them, we propose to take position relations into consideration and evaluate local patch correlations as well as keep the global expressions. When considering the general style transfer, Zhang et al. [51] suggested dealing with the content and style separately. While transferring the style, we also expect to preserve the original content of the image. The requirement of content fidelity naturally leads to the rise of the sparse representation-based style transfer. Sparse coding is an efficient way to model and preserve the image structure during reconstruction, and thus it is utilized by many methods as an effective tool to preserve image main contents. Sparse representation [7] is defined from the phenomenon that natural signals can be represented as a linear combination of a set of predefined atom signals. Based on this principle, image sparse representation is proposed and has been a rising area for image processing. Traditional image sparse representation [33, 48] assumed that relevant patches shared the same sparse representation. But transfers between styles, such as photo to sketch, reveal distinct characteristics cross domains. Instead of sharing the same representation, Wang et al. [43] then proposed a semi-coupled dictionary learning method to advance a linear cross-style mapping between related domains. However, the dictionary learning process of these

methods is based on a paired training set. In our case, many styles are *one-shot* cases or called *unsupervised style transfer*, namely that we have only one reference image without any explicitly defined correspondence. Thus, the paired training set is unavailable. Rosales et al. [35] tried to solve this issue via using a Bayesian technique for inferring the unknown mapping between the images to solve this unsupervised problem. In our method, we utilize a simple but very effective approach that exploits the edge features to build mappings for dictionary learning.

3 Joint-domain portrait oil painting stylization method

Some typical features make great art works unique and represent the style of a certain artist. To model those styles, we analyze amounts of oil portrait paintings. Through the analysis, three key factors appear to play key roles in affecting art rendering and visual feeling, i.e. the color, the structure and the texture. To transfer the desirable styles of the reference portrait oil painting to the input, we propose to decompose the painting into the features of the following three domains and transfer them jointly.

- **Color.** When looking at an oil painting, the first visual impact is generally created by the impression of colors. Artists utilize different pigments to distinguish various elements in the painting and express underlying emotions.
- **Structure.** The appearance of the painting highly relies on the basic structure of the frame. With the maintained structures, no matter what kind of artistic treatments is appended afterwards, the main contents stay stable.
- **Texture.** Coarse-to-fine textures interpret underlying contents. The stacking of strokes and the concealed pattern of the drawing surface combine an unique and significant characteristic of the drawing style.

In the proposed method, simulating the process of the artistic creation, the input is stylized via transferring the features in these domains jointly. The framework of the proposed stylization method is illustrated in Fig. 2 and more details can be viewed in the following sections.

3.1 Swatch-based color adjustment

To differentiate distinct elements in the frame, artists decide which pigment to use before stroking. The transfer in the color domain is, therefore, the first operation to be applied to the input portrait to adjust the color style of the input to get closer to the reference's tone.

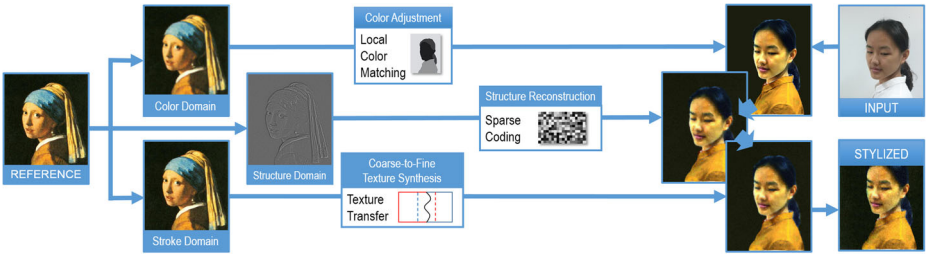


Fig. 2 Framework of the proposed joint-domain portrait oil painting stylization algorithm

There are many cross-channel correlations in common color spaces, such as RGB and HSV. These correlations may lead to distortions when modifying colors coherently [37]. Owing to this, we utilize the $l\alpha\beta$ color space [34], an orthogonal color space [10] for color adjustment to avoid distortions. As (1) showed, the image is transformed from the RGB color space to the $l\alpha\beta$ color space through the LMS cone space.

$$\begin{bmatrix} l \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} \log L \\ \log M \\ \log S \end{bmatrix},$$
$$\begin{bmatrix} L \\ M \\ S \end{bmatrix} = \begin{bmatrix} 0.3811 & 0.5783 & 0.0402 \\ 0.1967 & 0.7244 & 0.0782 \\ 0.0241 & 0.1288 & 0.8444 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}. \tag{1}$$

As shown in Fig. 3, the global color transfer method does not work well in this portrait stylization scenario, especially when the person is staying in a complex background in the input. To remedy this issue, we propose a local color transfer method via $l\alpha\beta$ color space.

In order to deal with local regions respectively, segmentation is needed in the first place. Hence, GrabCut [36] is conducted to find a binary mask that is refined by Matting Laplacian [22], and then the figure is distinguished from the background. Besides the background, there are usually three main semantic components of the figure from a common portrait: the hair, the face and the clothing. These three components also have to be modified separately to obtain a better-colorized result. However, it is really difficult to do partitions directly during color transfer without destroying the details. To solve this problem, we utilize the template obtained by the nonparametric representation [39] to detect 66 facial landmarks. With these landmarks and the binary mask, position relations are provided to obtain three swatches relating to the three regions automatically in Fig. 4. Then considering those swatches as cluster centers, the figure in the portrait clusters into three regions evaluated by the normalized feature vector $\mathbf{f} = \{R, G, B, \alpha, \beta\}$ as (2). In the feature vector $\mathbf{f}$, $\{R, G, B\}$ features are utilized to distinguish different colors while $\{\alpha, \beta\}$ features play a role in avoiding abnormities caused by luminance.

$$\min\left(\sum_{x,y}(\mathbf{f}^p_{x,y} - \mathbf{f}^s_{\mathbf{w}_i})^T \Gamma (\mathbf{f}^p_{x,y} - \mathbf{f}^s_{\mathbf{w}_i})\right), \tag{2}$$

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

Fig. 3 An example of color transfer. **a** Input image, **b** Reference image, **c** Global transfer result, **d** Proposed swatch-based transfer result

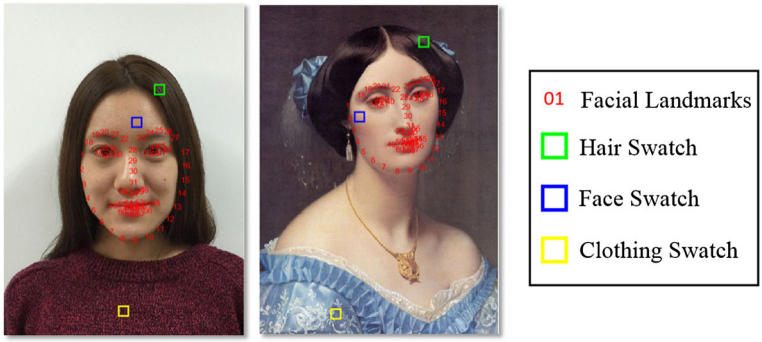


Fig. 4 The swatches in the input and the reference are obtained based on the binary mask and facial landmarks. Different colors are used to represent different swatch pairs

where $\mathbf{w}_i$ defines the center pixel position of the i-th swatch and pixel (x, y) belongs to the corresponding region Ω_i. $\mathbf{f}^p_{x,y}$ is the feature vector of pixel (x, y) while $\mathbf{f}^s_{\mathbf{w}_i}$ is the mean feature vector of the i-th swatch. Γ controls the balancing weight, which is set as $[0.001, 0.001, 0.001, 1, 1]$. According to this clustering, the portrait is segmented into three regions besides the background: the hair, the face and the clothing.

Since relevant regions have to be mapped between the input I^S and the reference I^T to transfer the corresponding color styles, the segmentation is executed on both of the images. Then, with statistics of three cluster pairs, the color of each pixel $I^S_{x,y}$ is shifted in the $l\alpha\beta$ color space depending on its nearest cluster centers.

$$I^C_{x,y} = \left[I^S_{x,y} - \bar{I}^S_{\Omega_i}\right] \times \frac{\sigma_{I^T_{\Omega_i}}}{\sigma_{I^S_{\Omega_i}}} + \bar{I}^T_{\Omega_i}, \quad (x, y) \in \Omega_i. \tag{3}$$

$I^C_{x,y}$ defines the color of each pixel in the colorized image I^C. $\bar{I}^S_{\Omega_i}$ and $\bar{I}^T_{\Omega_i}$ are respectively the mean of the i-th region Ω_i in the input I^S and the reference I^T while $\sigma_{I^S_{\Omega_i}}$ and $\sigma_{I^T_{\Omega_i}}$ indicate the standard deviations. Then three channels l, α and β are adjusted separately as (3). This transfer may still cause artifacts in local areas which changes dramatically in the neighborhood. To avoid artifacts, some extra detailed operations including smoothing images and flexibly changing weights of different dimensions of features to calculate the distances are conducted to yield the final colorized image I^C. In fact, the proposed local method works better than the global one in Fig. 3.

3.2 Structure reconstruction via sparse representation

With the desirable colors, artists paint the fundamental structure of the painting to determine the basic layout. The features in the structure domain are utilized here to maintain the structure of the input portrait.

In fact, image sparse representation, which refers that a patch can be approximately expressed as a linear combination of few predefined atom patches, is an efficient way to keep the image structure during reconstruction. Formally, the basic sparse model lets a column signal $g \in \mathbb{R}^n$ be described by a dictionary $D \in \mathbb{R}^{n \times m}$ in the following approximation problem:

$$g \approx D\gamma, \quad s.t. \ \|\gamma\|_0 \leq \psi, \tag{4}$$

where γ is the sparse representation of g and ψ refers to a predefined threshold. The l_0-norm $\|\cdot\|_0$ counts the nonzero elements of a vector and claims the sparsity of g. The dictionary D is trained as follows

$$D = \arg\min_{D,\alpha} \|\Lambda - D\alpha\|_2^2 + \lambda\|\alpha\|_0, \quad s.t. \ \ \|d_j\|_2^2 \leq 1, \forall j, \tag{5}$$

where d_j is the j-th dictionary atom of the dictionary D. While α is the sparse coefficient, Λ is a set of paired training patch samples. However, there is only one unique reference image in most cases, which leads to the inaccessibility of the paired training set. Therefore, we need to build mappings in the structure domain between the input and the reference for dictionary learning.

As the input and the reference describes different people, the cross-style mappings cannot be built directly. The edge feature [3], which represents structure information, is style-invariant upon most occasions and manipulated to relate the corresponding patches together and build the training set. We take advantage of the pyramid to offer edge features. To be more specific, the pyramid is constructed by filtering the image with a set of linear filters. The i-th subband B_i^C and B_i^T of the colorized input I^C and the reference I^T are calculated by the i-th linear filter in (6). The Haar filter is conducted in this paper.

$$B_i^T = f_i \otimes I^T, \quad B_i^C = f_i \otimes I^C. \tag{6}$$

The established pyramid with n levels has three subbands at each level, and there leaves a low-pass residue B_{3n+1}^T. We notice that subtracting the residue from the original image produces the edge map $(I^T - B_{3n+1}^T)$ so that edge patches that are ready for matching are isolated in Fig. 5.

Edge patches e^C and e^T from the input edge map $(I^C - B_{3n+1}^C)$ and the reference edge map $(I^T - B_{3n+1}^T)$ are then matched together based on the patch similarity $\delta(e^C, e^T)$. It is evaluated by both the intensity similarity and the structure similarity.

$$\delta(e^C, e^T) = \|e^C - e^T\|_2^2 + \tau\|\nabla e^C - \nabla e^T\|_2^2, \tag{7}$$

where τ defines a weighting parameter and set as 0.5 in this paper. ∇ is the gradient operator.

To summarize, with the guidance of edge features, the paired training set $\{C, T\}$ is acquired. Then inspired by [43], sparse coding is managed to learn the ultimate mapping relations. Traditional sparse representation based methods assume that the coupled dictionaries D^C and D^T strictly share the same sparse coefficients α for each patch pair. However, this assumption is too strong for cross-style patches, and we loose the assumption to admit that there exists a stable linear mapping W between the corresponding sparse coefficients

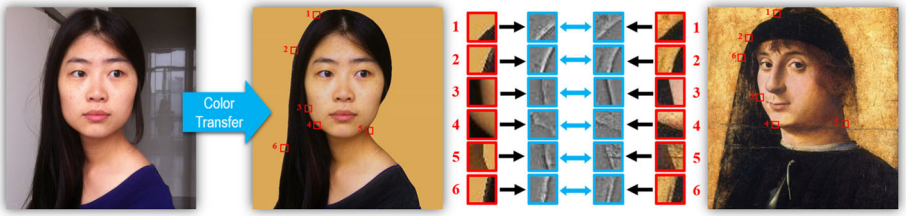


Fig. 5 Edge features are used to map similar patches between different styles for dictionary learning

α^C and α^T. The sparse dictionary learning problem is formulated as the following ridge regression problem in (8).

$$\begin{aligned}&\min_{\{D^C, D^T, W\}} \|C - D^C\alpha^C\|_F^2 + \|T - D^T\alpha^T\|_F^2 \\ &+\varphi\|\alpha^T - W\alpha^C\|_F^2 + \lambda^C\|\alpha^C\|_1 + \lambda^T\|\alpha^T\|_1 + \lambda^W\|W\|_F^2, \\ &s.t.\ \ \|d_j^C\|_2^2 \le 1, \|d_j^T\|_2^2 \le 1, \forall j,\end{aligned} \tag{8}$$

where d_j^C and d_j^T are the j-th dictionary atoms of the coupled dictionaries D^C and D^T. φ, λ^C, λ^T and λ^W refer to regularization parameters to balance different terms. $\|\cdot\|_F$ is the Frobenius-norm. Then the cross-style mapping W and the coupled dictionaries D^C and D^T are learned iteratively to optimize the variables alternatively.

Moreover, the image can be reconstructed with the learned dictionary. Taking k-th patch p_k^C in the colorized input portrait I^C as an example, the corresponding patch p_k^R in the structure-maintained output I^R can be reconstructed through the following optimization equation. The parameters φ, λ^C and λ^T are correspondingly set as 0.05, 0.01 and 0.001.

$$\begin{aligned}&\min_{\{\alpha_k^C, \alpha_k^R\}} \|p_k^C - D^C\alpha_k^C\|_F^2 + \|p_k^R - D^T\alpha_k^R\|_F^2 \\ &+\varphi\|\alpha_k^R - W\alpha_k^C\|_F^2 + \lambda^C\|\alpha_k^C\|_1 + \lambda^T\|\alpha_k^R\|_1.\end{aligned} \tag{9}$$

To solve the equation, we first sparsely code patch p_k^C on the dictionary D^C to obtain sparse coefficients α_k^C. Then patch p_k^R is initialized by enforcing sparse coefficients α_k^R with the mapping W to the reference dictionary D^T as $D^T W\alpha_k^C$. In the end, the structure-maintained output image I^R is predicted by the traversal and overlap of all patches p_k^R as follows

$$p_k^R = D^T\alpha_k^R. \tag{10}$$

3.3 Coarse-to-fine texture synthesis

After settling down the fundamental structures of the painting, artists draw with various brushes and creative strokes. Therefore, we analyze the features in the texture domain of the reference to accomplish the stroke texture synthesis in the structure-reconstructed input.

In this paper, we tend to manipulate the input portrait by patches in raster scanning. For each input patch, we search for a set of suitable reference patches satisfying the following two criterions:

- **Criterion 1:** The patches extracted from the reference should match the input patch to keep the original contents stable.
- **Criterion 2:** The extracted patches should not be too far from the input patch spatially, because the input and the reference originally offer the similar layouts.

When the k-th input patch p_k^R centered at (x, y) from the structure-reconstructed image I^R is traversing the reference I^T for potential candidate patches p_m^T, these two criterions work within some tolerance ζ to obtain a candidate patch set in (11).

$$\|p_k^R - p_m^T\|_2^2 + \phi\|\mathbf{u}_k^R - \mathbf{u}_m^T\|_2^2 < \zeta, \tag{11}$$

where ϕ controls the balance between the intensity similarity and the normalized distance. $\mathbf{u}_k^R$ and $\mathbf{u}_m^T$ refer to the center position vectors of the k-th patch p_k^R in the input image I^R and the m-th patch p_m^T in the reference image I^T.

With the candidate patch set, we seek a candidate patch which can fit in with its neighbors best to paste into the result. To fit in seamlessly, the appropriate patch $p^T_{m'}$ for synthesizing patch p^E_k that centers at pixel (x, y) in the stroke-synthesized image I^E should minimize the cumulative minimum error $Q_{x,y}$ after some cuts. Along the vertical edge of the synthesized patch p^E_{k-1} and the candidate patch $p^T_{m'}$, the overlap regions are defined as O^E_{k-1} and $O^T_{m'}$ respectively. The chosen patches with jagged edges compose the stroke texture features I^E of the input subsequently by dynamic programming.

$$\begin{aligned} Q_{x,y} &= \min(Q_{x-1,y-1}, Q_{x-1,y}, Q_{x-1,y+1}) + \|O^E_{k-1} - O^T_{m'}\|^2_2, \\ s.t.\ \ I^E &= \arg\min_{I^E} Q. \end{aligned} \tag{12}$$

Then the simulated image with stroke textures I^E is blended with the structure-maintained image I^R through alpha matting [32]. The stroke-confluent result I^K is synthesized under the restraint of the predefined weighting parameter η as 0.9.

$$I^K = \eta I^E + (1 - \eta) I^R. \tag{13}$$

At the same time, it is noticed that when artists are drawing with brushes on the drawing surface, there exist some fine-grained textures from the surface which cannot be completely covered by strokes. Enlightened by Sunkavalli's work [41], these fine-grained textures are regarded as noises and imitated by pyramid matching. After pyramid matching, it comes out that the fine-grained textures induced by the surface have been injected to the input. We include pseudo-code of the overall joint domain portrait stylization algorithm in Algorithm 1.

Algorithm 1 Joint Domain Portrait Stylization

Input: Input image I^S, reference style I^T
Output: Stylized image I^K

1: $\Omega \leftarrow$ SwatchBasedClustering(I^S, I^T) ((2))
2: $I^C \leftarrow$ ColorAdjustment(I^S, I^T, Ω) ((3))
3: $I^R \leftarrow$ StructureReconstruction(I^C, I^T) ((8)-(10))
4: $I^E \leftarrow$ TextureSynthesis(I^R, I^T) ((12))
5: $I^K \leftarrow$ TextureStrctureBlending(I^E, I^R) ((13))

4 Experimental results

To evaluate the effectiveness of the proposed method, we conduct experiments on several test image pairs. In experiments, we import an original portrait photo as the input to be stylized and a portrait oil painting as the reference. The reference portrait oil painting set is collected from the Internet. We also take some portrait photos by iPhone 5 as the input to test the algorithm. These corresponding pairs are downsampled into the same size, around 300,000 pixels. The patch size is 5×5.

In the meanwhile, the background also plays an important role in the mood of a portrait. But it turns out that artists usually design the background instead of painting the authentic one in real life when drawing portraits. Thus in experiments, the previously obtained mask is utilized to extract the background and directly replace the background of the synthetic

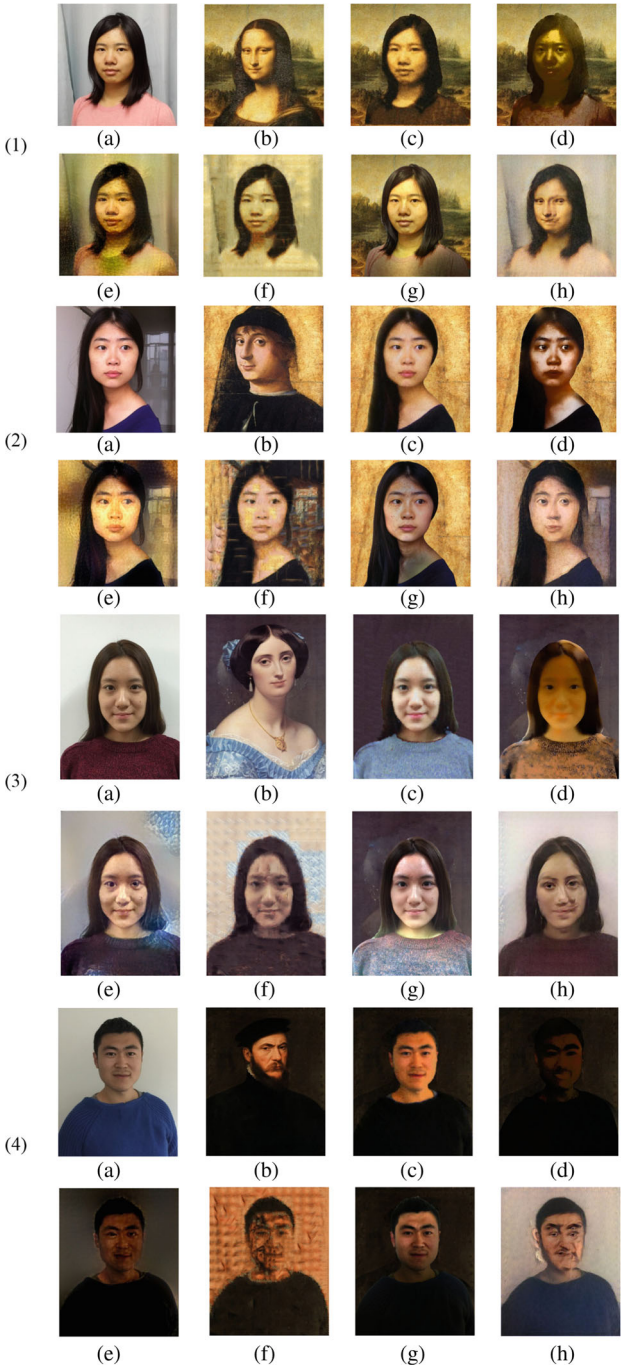


Fig. 6 Subjective experimental results. **a** Original input stylish image. **b** Reference image. **c** Stylized image by the proposed method. **d** Stylized image by Sunkavalli's method [41]. **e** Stylized image by Gatys' method [12]. **f** Stylized image by Frigo's method [11]. **g** Stylized image by Shih's method [40]. **h** Stylized image by Li's method [23]

Table 1 Scores of different methods

Images	Proposed	Sunkavalli's	Gatys'
Image #1	4.38	2.00	3.18
Image #2	3.95	2.95	3.35
Image #3	3.73	1.60	2.85
Image #4	3.60	2.08	3.18
Average	3.92	2.16	3.14

portrait through Poisson image editing [31]. When the reference mask and the input mask cannot overlap perfectly, image inpainting [14] is utilized to extrapolate the missing area.

We compare the proposed algorithm with methods in two categories. One transforms the input into the reference style, such as Sunkavalli's method [41], Gatys' method [12], Frigo's method [11], Shih's method [40], and Li's method [23]. The other one only stylizes the input into a predefined oil painting style, instead of a customized style, such as Zhao's method [54] and the oil painting filter of the *Glaze* app [6] and the *Meitu* app [29]. Then subjective results are illustrated in Figs. 6 and 8. In the meanwhile, to ensure the credibility of our method, we invited 40 observers with diversity to fill in our questionnaires. The number of female observers is 20 while the age varies from 18 to 63. The distinct professions are computer, education, design, finance and so on. To verify the fairness, the orders of results randomly change every round. The statistics of the survey are shown in Table 1, Fig. 7a and b.

4.1 Transfer with reference

When comparing with methods which focus on transferring a customized reference style, stylization results are exhibited in Fig. 6. Our method transforms the reference style to the input containing colors and coarse-to-fine textures while preserving the original structures. Sunkavalli's method [41] harmonizes the figure into the reference seamlessly, but also leads to the darkness of the figure and the loss of some details. In the meanwhile, neural networks based method [12] works sensitively to conspicuous patterns but is inefficient on relatively

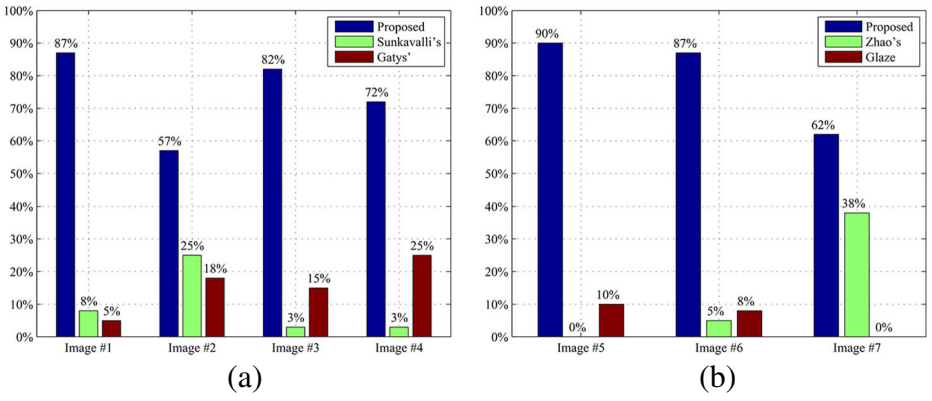


Fig. 7 User study. **a** Statistics of votes for the method which produces the most similar style. **b** Statistics of votes for the method which produces the most favorable oil painting

smooth oil paintings. Shih's method [40] only transfers colors without simulating the brush stroke of oil paintings. Frigo's method [11] creates many repeated textures and Li's method [23] loses the original facial features.

In the survey, observers are asked to score the similarity of styles between the reference and the stylized input from 5 to 1, from the most similar one to the least similar one. According to the statistics in Table 1, our method acquires the highest score in each round of Fig. 6. We also ask observers to choose the best method which transforms the most similar style in every round. 75% observers in average agree that our method works the best in Fig. 7a.

4.2 Transfer without reference

At the same time, we compare our method with predefined style transfer methods in Fig. 8. Our method transfers the reference style to the input accompanied with subtle esthetic pleasures. Zhao's method [54] simulates the process of brush stroking efficiently, but the coloring is undesirable. As for the *Glaze* app and the *Meitu* app, it is a filter, and the results seem rather artificial.

In this category, we invite observers to choose their favorite stylized image in each round of Fig. 8. We observe in Fig. 7b that more than 60% observers love our method the most in each round.

From the above two points of view, our method both performs well. It signifies that the proposed method produces fairly similar and visually desirable stylized images.

4.3 Parameter analysis

To verify the robustness of our method, we analyze the parameter settings. In (9) and (8), there are three regularization parameters to balance different terms: φ, λ^C and λ^T. In the

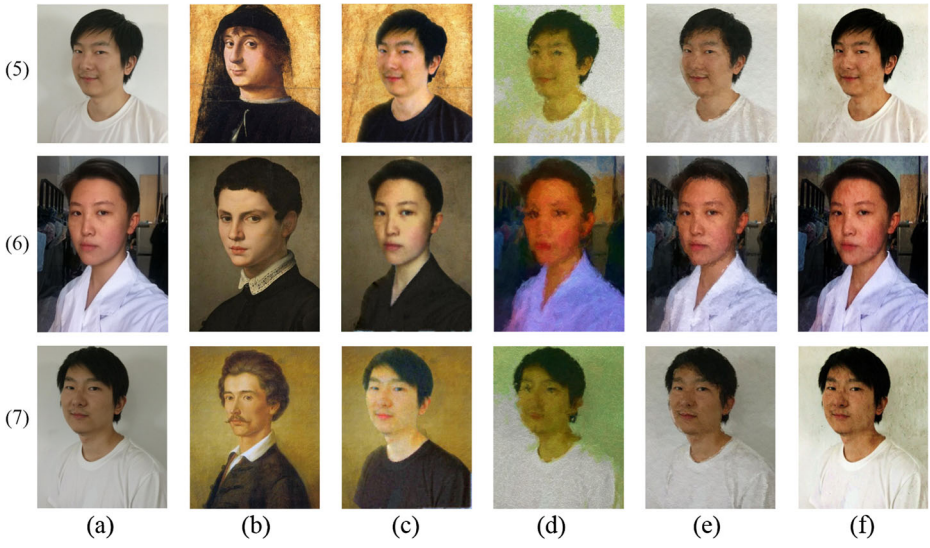


Fig. 8 Subjective experimental results. **a** Original input stylish image. **b** Reference image. **c** Stylized images by the proposed method. **d** Stylized image by Zhao's method [54]. **e** Stylized image by *Glaze* app [6]. **e** Stylized image by *Meitu* app [29]

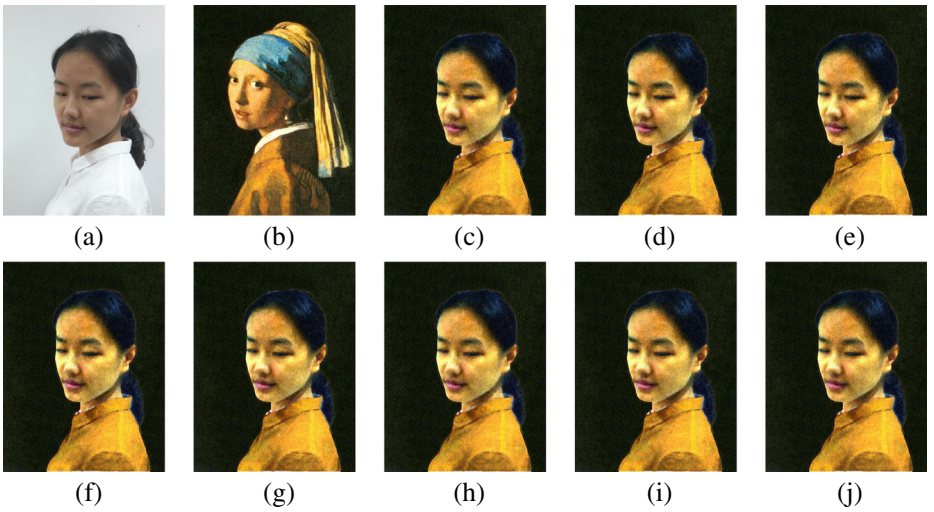


Fig. 9 Effects of parameters φ, λ^C and λ^T. The visual difference is hardly noticeable, indicating the robustness of our method. **c**-**j** Stylization results using different parameter settings of (φ, λ^C, λ^T). **c** (φ, λ^C, λ^T) = (0.05, 0.01, 0.001). **d** (0.01, 0.01, 0.001). **e** (0.1, 0.01, 0.001). **f** (0.05, 0.1, 0.001). **g** (0.05, 0.001, 0.001). **h** (0.05, 0.01, 0.01). **i** (0.05, 0.01, 0.0001). **j** (0.05, 0.001, 0.01)

experiment, the parameters are set in different values but the subjective visual perceptions have not changed much as shown in Fig. 9. We set φ, λ^C and λ^T as 0.05, 0.01 and 0.001 empirically in this paper.

4.4 Limitations

While our method works on a lot of input pairs, the style transfer may fail in some cases because the input and reference have very different appearances. For instance, the face detector is unable to locate the landmarks in the side face photo. In Fig. 10, due to the original input image is a side image, we cannot locate the landmarks accurately which leads to the mismatch of corresponding regions. The color transfer result in Fig. 10c is obtained

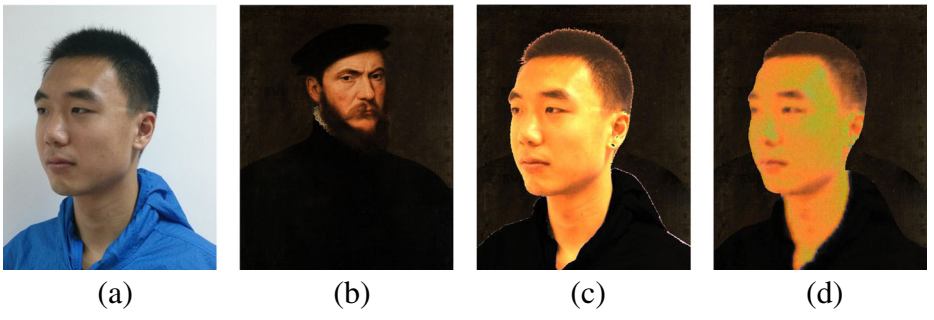


Fig. 10 An example of failure cases. **a** Original input stylish image. **b** Reference image. **c** Color transfer result. **d** Stylized result

while the hair region in Fig. 10a is mismatched with the face region in Fig. 10b and results in the bad performances of the proposed method in Fig. 10d. At the same time, region matching may fail when the input photo has very poor illuminations.

5 Conclusions

In this paper, we propose a joint-domain method to stylize portraits into a customized oil painting style automatically. We analyze three factors to evaluate the style of the portrait oil painting, the color, the structure, and the texture. Each feature plays an important role in the expression of the painting and is applied to the input in order to imitate the reference's style. Hence, the color style and coarse-to-fine textures are blended to the input while the structure of the input maintains. Experimental results indicate that our proposed portrait oil painter outperforms state-of-art algorithms in most people's eyes.

There are some interesting issues for the future work. A direction for future work is the style transfer from multiple references. It might be possible to stylizing different face regions in the input photo from different oil paintings to better match the input face and achieve more flexibility. Besides, we will investigate the automatic selection of the best matched oil painting given input photos. The selection could be achieved by leveraging deep neural networks to extract high-level semantic features. We believe both investigation directions will solve the problem of mismatches between the input photo and the oil painting.

Acknowledgments This work was supported by the National Natural Science Foundation of China under Contract 61472011.

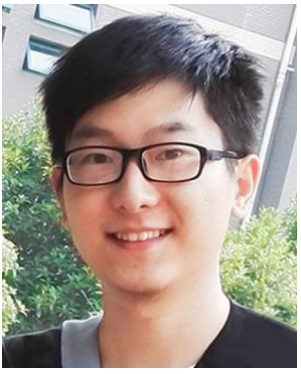
References

1. Ashikhmin M (2003) Fast texture transfer. IEEE Comput Graph Appl 23(4):38–43
2. Berger I, Shamir A, Mahler M, Carter E, Hodgins J (2013) Style and abstraction in portrait sketching. ACM Trans Graph 32(4):96–96
3. Bhujle H, Chaudhuri S (2014) Novel speed-up strategies for non-local means denoising with patch and edge patch based dictionaries. IEEE Trans Image Process 23(1):356–365
4. Bousseau A, Kaplan M, Thollot J, Sillion F (1983) Interactive watercolor rendering with temporal coherence and abstraction. In: International Symposium on Non-photorealistic Animation and Rendering, pp 141–149
5. Curtis CJ, Anderson SE, Seims JE, Fleischer KW, Salesin D (1997) Computer-generated watercolor. In: Conference on Computer Graphics and Interactive Techniques, pp 421–430
6. Dezeustre G (2014) Glaze app https://itunes.apple.com/us/app/glaze/id521573656
7. Donoho DL (2006) Compressed sensing. IEEE Trans Inf Theory 52(4):1289–1306
8. Efros A, Leung TK et al (1999) Texture synthesis by non-parametric sampling. In: Proceedings of the seventh IEEE International Conference on Computer Vision, vol 2. IEEE, pp 1033–1038
9. Efros AA, Freeman WT (2001) Image quilting for texture synthesis and transfer. In: Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques. ACM, pp 341–346
10. Flanagan P, Cavanagh P, Favreau OE (1990) Independent orientation-selective mechanisms for the cardinal directions of colour space. Vis Res 30(5):769–778
11. Frigo O, Sabater N, Delon J, Hellier P (2016) Split and match: example-based adaptive patch sampling for unsupervised style transfer. In: IEEE Conference on Computer Vision and Pattern Recognition, pp 553–561
12. Gatys LA, Ecker AS, Bethge M (2016) Image style transfer using convolutional neural networks. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
13. Greenfield GR, House DH (2003) Image recoloring induced by palette color associations. Journal of Wscg 11:189–196
14. He K, Sun J (2014) Image completion approaches using the statistics of similar patches. IEEE Trans Pattern Anal Mach Intell 36(12):2423–2435

15. Hertzmann A (1998) Painterly rendering with curved brush strokes of multiple sizes. In: Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques. ACM, pp 453–460
16. Hertzmann A, Jacobs CE, Oliver N, Curless B, Salesin D (2001) Image analogies. In: Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques. ACM, pp 327–340
17. Huang TW, Chen HT (2009) Landmark-based sparse color representations for color transfer. In: IEEE 12Th International Conference on Computer Vision. IEEE, pp 199–204
18. Jia K, Wang X, Tang X (2013) Image transformation based on learning dictionaries across image spaces. IEEE Trans Pattern Anal Mach Intell 35(2):367–380
19. Kyprianidis JE, Collomosse J, Wang T, Isenberg T (2013) State of the art: a taxonomy of artistic stylization techniques for images and video. IEEE Trans Vis Comput Graph 19(5):866–885
20. Lee H, Seo S, Ryoo S, Yoon K (2010) Directional texture transfer. In: Proceedings of the 8th International Symposium on Non-Photorealistic Animation and Rendering. ACM, pp 43–48
21. Levin A, Lischinski D, Weiss Y (2004) Colorization using optimization. ACM Trans Graph 23(3):689–694
22. Levin A, Lischinski D, Weiss Y (2008) A closed-form solution to natural image matting. IEEE Trans Pattern Anal Mach Intell 30(2):228–242
23. Li C, Wand M (2016) Combining markov random fields and convolutional neural networks for image synthesis. In: IEEE Conference on Computer Vision and Pattern Recognition, pp 2479–2486
24. Liu Y, Zheng Y, Liang Y, Liu S, Rosenblum DS (2016) Urban water quality prediction based on multi-task multi-view learning. In: Proceedings of the International Joint Conference on Artificial Intelligence
25. Liu X, Ma L, Liu Y (2017) Global tone: using tone to draw in pen-and-ink illustration. Multimed Tool Appl 76(10):12853–12869
26. Lo KH, Wang YCF, Hua KL (2016) Example-based image textural style transfer. IEEE Multimed 23(4):60–66
27. Lu Y, Wei Y, Liu L, Zhong J, Sun L, Liu Y (2017) Towards unsupervised physical activity recognition using smartphone accelerometers. Multimed Tool Appl 76(8):10,701–10,719
28. Marr D (1982) Vision: a computational investigation into the human representation and processing of visual information. Henry Holt and Co., Inc., New York
29. Meitu app. https://itunes.apple.com/cn/app/id463422433
30. Paget R, Longstaff D (1995) Texture synthesis via a non-parametric markov random field. In: Proceedings of DICTA-95, Digital Image Computing: Techniques and Applications, vol 1, pp 547–552
31. Pérez P, Gangnet M, Blake A (2003) Poisson image editing. ACM Trans Graph 22(3):313–318
32. Porter T, Duff T (1984) Compositing digital images. ACM Siggraph Comput Graph 18(3):253–259
33. Protter M, Elad M (2009) Image sequence denoising via sparse and redundant representations. IEEE Trans Image Process 18(1):27–35
34. Reinhard E, Ashikhmin M, Gooch B, Shirley P (2001) Color transfer between images. IEEE Comput Graph Appl 21(5):34–41
35. Rosales R, Achan K, Frey B (2003) Unsupervised image translation. In: Proceedings of IEEE International Conference on Computer Vision, pp 472–472
36. Rother C, Kolmogorov V, Blake A (2004) Grabcut: interactive foreground extraction using iterated graph cuts. ACM Trans Graph 23(3):309–314
37. Ruderman DL, Cronin TW, Chiao CC (1998) Statistics of cone responses to natural images: implications for visual coding. J Opt Soc Am 15(8):2036–2045
38. Salisbury M, Anderson CR, Lischinski D, Salesin D (1996) Scale-dependent reproduction of pen-and-ink illustrations. Br J Sports Med 48(7):622
39. Saragih JM, Lucey S, Cohn JF (2009) Face alignment through subspace constrained mean-shifts. In: IEEE 12th International Conference on Computer Vision. IEEE, pp 1034–1041
40. Shih YC, Paris S, Barnes C, Freeman WT, Durand F (2014) Style transfer for headshot portraits. ACM Trans Graph 33(4):1–14
41. Sunkavalli K, Johnson MK, Matusik W, Pfister H (2010) Multi-scale image harmonization. ACM Trans Graph 29(4):125
42. Wang X, Tang X (2009) Face photo-sketch synthesis and recognition. IEEE Trans Pattern Anal Mach Intell 31(11):1955–1967
43. Wang S, Zhang L, Liang Y, Pan Q (2012) Semi-coupled dictionary learning with applications to image super-resolution and photo-sketch synthesis. In: IEEE Conference on Computer Vision and Pattern Recognition. IEEE, pp 2216–2223
44. Wang T, Collomosse J, Hunter A, Greig D (2013) Learnable stroke models for example-based portrait painting. In: British Machine Vision Conference, pp 36.1–36.11
45. Wang Q, Chen D, Li S, Wu Q, Zhang Q (2017) An adaptive cartoon-like stylization for color video in real time. Multimed Tool Appl 76(15):16767–16782

46. Welsh T, Ashikhmin M, Mueller K (2002) Transferring color to greyscale images. ACM Trans Graph 21(3):277–280
47. Winkenbach GA, Salesin D (1994) Computer-generated pen-and-ink illustration. In: Conference on Computer Graphics and Interactive Techniques, pp 91–100
48. Yang J, Wright J, Huang TS, Ma Y (2010) Image super-resolution via sparse representation. IEEE Trans Image Process 19(11):2861–2873
49. Yang L, Chen WB, Zhang C, Johnstone JK, Gao S, Warner G (2012) Profiling online auction sellers using image-editing styles. IEEE Multimed 19(1):29–29
50. Yang Y, Zhao H, You L, Tu R, Wu X, Jin X (2017) Semantic portrait color transfer with internet images. Multimed Tool Appl 76:523–541
51. Zhang W, Cao C, Chen S, Liu J (2013) Style transfer via image component analysis. IEEE Trans Multimed 15(7):1594–1601
52. Zhao M, Zhu SC (2010) Sisley the abstract painter. In: Proceedings of the 8th International Symposium on Non-Photorealistic Animation and Rendering. ACM, pp 99–107
53. Zhao M, Zhu SC (2011) Portrait painting using active templates. In: International Symposium on Non-Photorealistic Animation and Rendering 2009, Vancouver, Bc, Canada, August, pp 117–124
54. Zhao M, Zhu SC (2013) Abstract painting with interactive control of perceptual entropy. ACM Transactions on Applied Perception 10(1):5

Saboya Yang received the B.S. degree in computer science from Peking University, Beijing, China, in 2014, where she is currently pursuing the Master degree with the Institute of Computer Science and Technology. Her current research interests include nonlocal means, sparse representation and image stylization.



Shuai Yang received the B.S. degree in computer science from Peking University, Beijing, China, in 2015, where he is currently pursuing the Ph.D. degree with the Institute of Computer Science and Technology. His current research interests include image inpainting, depth map enhancement and image stylization.

Wenhan Yang received the B.S degree in computer science from Peking University, Beijing, China, in 2012, where he is currently pursuing the Ph.D. degree with the Institute of Computer Science and Technology. He was a Visiting Scholar with the National University of Singapore, Singapore, from 2015 to 2016. His current research interests include image processing, sparse representation, image restoration and deep learning-based image processing.

Jiaying Liu received the B.E. degree in computer science from Northwestern Polytechnic University, Xi'an, China, and the Ph.D. degree with the Best Graduate Honor in computer science from Peking University, Beijing, China, in 2005 and 2010, respectively. She is currently an Associate Professor with the Institute of Computer Science and Technology, Peking University. She has authored over 90 technical articles in refereed journals and proceedings, and holds 19 granted patents. Her current research interests include image/video processing, compression, and computer vision.

Dr. Liu was a Visiting Scholar with the University of Southern California, Los Angeles, from 2007 to 2008. She was a Visiting Researcher at Microsoft Research Asia (MSRA) in 2015 supported by Star Track for Young Faculties. She has also served as TC member in IEEE CAS MSA and APSIPA IVM, and APSIPA distinguished lecture in 2016–2017. She is CCF/IEEE Senior Member.