Photo Stylistic Brush: Robust Style Transfer via Superpixel-Based Bipartite Graph

Jiaying Liu, *Senior Member, IEEE,* Wenhan Yang, *Student Member, IEEE,* Xiaoyan Sun, *Senior Member, IEEE,* Wenjun Zeng, *Fellow, IEEE*

***Abstract*—With the rapid development of social network and multimedia technology, customized image and video stylization has been widely used for various social-media applications. In this paper, we explore the problem of exemplar-based photo style transfer, which provides a flexible and convenient way to invoke fantastic visual impression. Rather than investigating some fixed artistic patterns to represent certain styles as was done in some previous works, our work emphasizes styles related to a series of visual effects in the photograph, *e.g.* color, tone, and contrast. We propose a photo stylistic brush, an automatic robust style transfer approach based on Superpixel-based BIpartite Graph (SuperBIG). A two-step bipartite graph algorithm with different granularity levels is employed to aggregate pixels into superpixels and find their correspondences. In the first step, with the extracted hierarchical features, a bipartite graph is constructed to describe the content similarity for pixel partition to produce superpixels. In the second step, superpixels in the input/reference image are rematched to form a new superpixel-based bipartite graph, and superpixel-level correspondences are generated by a bipartite matching. Finally, the refined correspondence guides SuperBIG to perform the transformation in a decorrelated color space. Extensive experimental results demonstrate the effectiveness and robustness of the proposed method for transferring various styles of exemplar images, even for some challenging cases, such as night images.**

***Index Terms*—Image stylization, superpixel, bipartite graph, stylistic brush.**

I. INTRODUCTION

With the prevalence of multimedia social networking, it has become popular to share photos online. Most people nowadays prefer uploading photos with special artistic enhancement made by various Apps such as Facebook and Instagram instead of the original ones. This kind of photo style enhancement makes pictures dramatically more impressive and inspires new imagination. However, existing systems either allow users to only roughly change the photo in a fixed template, or require a series of subtle processes by experienced photographers using the editing software.

Image style transfer aims to automatically change the *stylistic elements* of an input image (color, texture, contrast, *etc.*) to follow a given exemplar, *e.g.* well-known paintings or fabulous pictures taken by professional photographers. Early works start by transferring one of these elements among images. The color transfer methods either extract the most representative colors from the images and build a conversion algorithm between those colors [1], [2], or directly adjust the color distribution via a histogram feature fitting [3], [4]. Contrast is usually transferred in the frequency band space, such as the bilateral space [5], Laplacian pyramid [6] or Haar pyramid [7]. Since these methods only consider one specific stylized element, they may produce some visual effect, but are difficult to be applied widely in practice.

Meanwhile, the image stylization is also explored in the computer graphics community, referred to as non-photorealistic rendering (NPR). It aims to generate non-photorealistic style images, such as watercolor painting [8], sketch generation [9] and abstract drawing [10]. By a carefully crafted design, a bunch of stylized elements are extracted to represent the artistic style of an image and further used to transfer artistic visual effects. However, these hand-crafted features, designed with certain type of artworks, lack expandability by nature and are not adaptive in representing other styles or new styles.

In real applications, it is unrealistic to ask most people to give a specific description about what style they exactly want. Usually, what they could offer is a real example they saw before, *e.g. Mona Lisa*, or an abstract word they read from books, *e.g. Baroque*. Knowing little about image editing, they need a tool to define a bunch of style settings from these examples and make adjustments automatically. Like the *format painter* of Microsoft Office, *Stylistic Brush* provides a desirable and powerful tool to enable an automatic arbitrary style transfer between images. The style is extracted dynamically from a fantasy *reference image* (also referred to as target image). A new output image is synthesized based on the content of the input image and the extracted styles of the reference one.

Therefore, some works investigating image stylization by considering the style composition instead of a single style element are emerging. Most of these methods devote to separating and dealing with the content and style individually. An early work [11] explored the concept of *image analogy* by building a multiscale autoregression framework to adaptively learn a wide variety of *image filter*. Zhang *et al.* [12] proposed

This work was supported by National Natural Science Foundation of China under Grant 61772043, Microsoft Research Asia project ID FY17-RES-THEME-013 and CCF-Tencent Open Research Fund. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Sen-Ching Samson Cheung. (*Corresponding author: Xiaoyan Sun.*)

Jiaying Liu and Wenhan Yang are with Institute of Computer Science and Technology, Peking University, Beijing, 100080, China, e-mail: {liujiaying, yangwenhan}@pku.edu.cn.

Xiaoyan Sun and Wenjun Zeng are with Microsoft Research Asia, Beijing, 100080, China, e-mail: {xysun, wezeng}@microsoft.com.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

to perform an image component analysis to decompose an image into three components and constructed a coarse-to-fine Markov random field to propagate colors in the paint and edge components. In [13], a deep network-based method was proposed to separate and recombine the content and style. A composition of the learned CNN features gives a clue of content correspondence and guides the production of new artistic images via transferring the style features.

These methods suffer from two limitations: 1) From the model aspect, the assumption that the content and the style could be separable may be questional. Some common observations, such as sunset with red color and grass with certain texture patterns, lead to the conclusion that some styles are highly correlated with the image content. Thus, previous methods with such a separable assumption lose some style information in the transformation. 2) From the application aspect, these methods mainly focus on painting styles and are good at transferring or generating texture styles. However, comparing with paintings, people usually pay more attention to visual effects caused by color, light, contrast etc. than textures in the photography, especially in the area of photo reference based image editing [1], [5], [14]–[18]. Since textures are treated as the property related to scenes and objects, instead of the individual style. More related detailed analysis about photography and human vision can be found in [19]–[22].

In this paper, we aim to create a stylistic brush to help people beautify their photos by transferring desirable styles of a chosen exemplar image to the input one. Focusing on photos, we pay more attention to the color, light and contrast of a photograph instead of the factors related to art, such as textures or strokes. Compared to previous methods, we make two more reliable assumptions: 1) For most photos, the Internet enables us to collect a content similar reference with a favorable style. It is usually the case for a certain category of images, such as the landmark or face images; 2) Different from general content-based features, we obtain matched points of the same scene between the reference and input images as more reliable guidance of content similarity, via dense correspondence detection methods.

With the above considerations, the proposed stylistic brush is realized by a robust style transfer method based on the Superpixel BIpartite Graph (SuperBIG) framework for image stylization. First, a dense correspondence between the input and reference images is estimated to obtain matched pixels as the primitives. By exploiting hierarchical features in different-granularity, we measure the distances from pixels to the identified matched points in the feature space to cluster these pixels into superpixels. Then a bipartite graph partition is exploited to assign uncluttered pixels into superpixels by considering both the local and global consistency. Afterwards, superpixels of two images are rematched to form a new superpixel bipartite graph to refine the final superpixel-level correspondent relationship. Finally, SuperBIG transfers colors within each superpixel correspondence in a decorrelated color space to achieve the stylization.

The main contributions of our work are summarized as follows:

- We propose an automatic robust style transfer framework based on the Superpixel BIpartite Graph (SuperBIG) as a *Stylistic Brush* for practical photo stylization. It uses a hierarchical abstraction scheme to integrate the local consistency of superpixel and the global adaptation between input and reference images. With any aligned inputs, SuperBIG also provides a refined superpixel-level correspondence for accurate and robust local style transfer.
- Benefiting from diversity of the proposed hierarchical features in different granularity, as well as the advantages of the unified bipartite graph framework, SuperBIG achieves promising results in terms of effectiveness and robustness in extensive experiments, even for some challenging cases, such as night images and scene change cases. Extensive experiments demonstrate that our method significantly outperforms previous methods in the general style transfer.

The remainder of this paper is organized as follows: Section II gives a brief overview of the related work. In Section III, we present how to create the superpixel correspondence between input and reference images by the proposed superpixel bipartite graph model with hierarchal features, and further apply it for an effective and robust style transfer. Experimental results are illustrated in Section IV. Section V briefly discusses our limitations and related future directions. Finally, concluding remarks are given in Section VI.

II. Related Works

A. Non-Photorealistic Rendering

Non-photorealistic rendering was first proposed by Winkenbach and Salesin [23]. It aims to produce images derived from a wide variety of styles such as painting, drawing, sketching, illustration and animation for digital art. Non-experts can transfer artistic styles of famous painters to ordinary photos taken everyday with the help of NPR. Nowadays, many ad-hoc NPR schemes have been proposed for this task with a varying degree of success [24]. While Li *et al.* [25] proposed to create and view interactive exploded views of 3D models, Pouli and Reinhard [17] utilized a user-specified target image's color palette to achieve creative effects. For artistic styles rendering, some researchers focus on simulating virtual brush strokes to obtain a particular style [26], [27]. Region-based methods are also used to independently render the interiors of regions [28], [29]. In the meantime, many image processing filters have been applied to produce images in artistic styles [30], [31]. Different from NPR studying on artistic patterns rendering, our work aims to address the challenge of photo style transfer, where more diversified styles are faced and photometric properties, such as light, contrast, change more abruptly within an image.

B. Hand-Crafted Style Transfer

Hand-crafted style transfer techniques aim to adjust the color, contrast and tone of images, with the aid of signal properties, *e.g.* the statistic information of colors, without

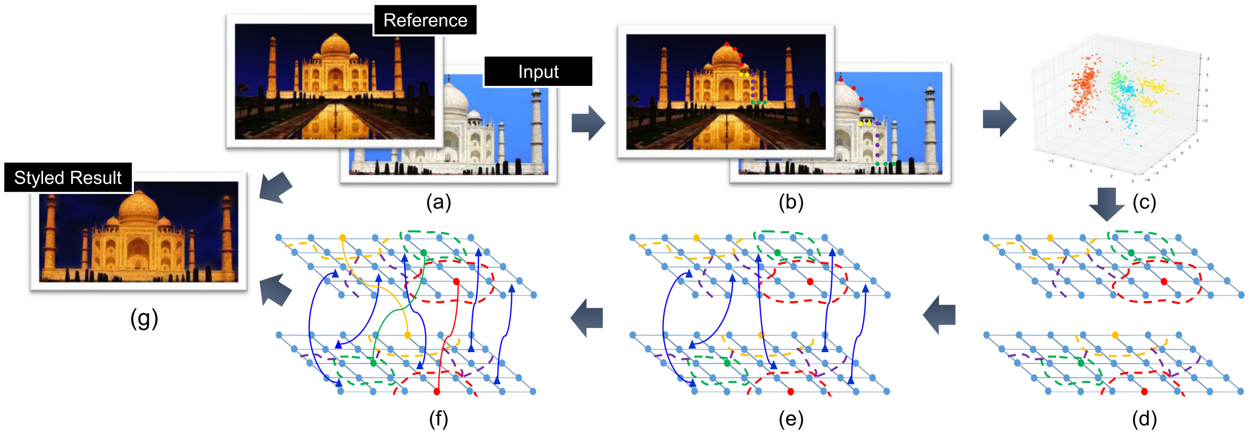


Fig. 1. The flowchart of SuperBIG algorithm. (a) Input and reference images. (b) Matched points detected by dense correspondence method. (c) Hierarchical features for each pixel. (d) Superpixels obtained by the distance between each pixel and matched points. (e) Superpixels obtained by pixel-level bipartite graph partition. (f) The superpixel correspondence generated by superpixel bipartite graph matching. (g) The styled result based on colors of input and reference images, as well as the superpixel correspondence.

considering the content-level correspondence. For color transfer, the work in [1] transferred colors by matching the statistics of color distributions. Subsequent works improved the accuracy and robustness of statistical estimation, such as soft-segmentation [32], multi-dimensional distribution matching [33] and minimal displacement mapping [17]. There are also some methods [34], [35] that consider colorizing the image with user defined colors. These methods propagate colors with an elaborately designed constraint to ensure natural visual effect of the produced result. For contrast and tone, adjustment is manipulated in the frequency domain, such as bilateral space [5], Laplacian pyramid [6] or Haar pyramid [7]. Our work focuses on transferring photo styles adaptively based on the given references instead of a crafted architecture designed for the transfer of a certain style.

C. Example-Based Style Transfer

For image stylization, exploiting only signal properties and statistical correspondence cannot capture the local region correspondence for the local style transfer. Recently, some methods explore ways to create and utilize the content-level correspondence to benefit the stylization. In [36], [37], the input and reference images are segmented first. Then, colors are propagated from color images to greyscale images via a set of locally homogeneous patches or basic elements called color scribbles. Charpiat *et al.* [38] assigned colors to the greyscale image by solving an optimization problem in the framework of graph cut. In [15], after manual segmentation of major foreground objects, a belief-propagation colorizes the greyscale image with the help of Internet images. In [15], [39], colors are transferred by estimating per-pixel registered correspondence between input and reference images. Kumar *et al.* [16] proposed to create correspondences between superpixels by fast cascade feature matching, and then refine the transfer results by a voting approach. Cheng *et al.* [14] proposed a superpixel-based recoloring scheme based on a soft matching embedded with color statistics, texture characteristics and spatial constraints to generate new recolored images. There are also some works that aim to conduct favorite exemplars recommendation based on visual information [14] or patch aggregation [40]. Several methods [41]–[43] focus on addressing the local style transfer on a specific category - the facial image with assumed face-related priors or by utilizing external coupled time-lapse videos to create the local transfer mapping. Compared with them, our method has distinguished properties and advantages. Generally, our method is not limited to portrait images and does not utilize extra information. Specifically, approaches in [41] and [43] create local transfer mapping based on the facial structures priors of portrait images. For example, facial landmarks are assumed given to provide location correspondences in [41]. In [42], the local mapping is built based on the style changes in the same location of a time-lapse video from the same view. In our method, without the aid of crafted priors or extra data, a hierarchical feature covering low- to high-level context is built to create the initialized local correspondences, then a two-step bipartite framework is utilized to refine the local transfer by jointly optimizing the partition and matching across the whole image to ensure the local and global consistency.

III. SUPERPIXEL BIPARTITE GRAPH FOR PHOTO STYLE TRANSFER

The proposed SuperBIG transfers the style of the reference image to the input image by a two-step bipartite graph framework as shown in Fig. 1. SuperBIG first detects the dense correspondence (Fig. 1(b)) and calculates the designed hierarchical features (Fig. 1(c)). Based on the correspondence and features, SuperBIG then aggregates pixels into superpixels using a simple clustering algorithm (Fig. 1(d)) for the pixels around the matched points and a bipartite graph framework (Fig. 1(e)) for the pixels far from the matched points. Afterwards, SuperBIG transfers the colors between corresponding superpixels (Fig. 1(f)) in a decorrelated color space.

A. Superpixel Aggregation with Hierarchical Features

Superpixel is a pixel cluster consisting of several pixels with similar color and brightness. It is proposed to well define coherent regions, as basic elements of over-segmentation. It usually provides an initialization for segmentation [44]–[46] or a soft constraint on segmentation [47], [48]. Compared with raw pixels, superpixel is a more sparse and efficient representation, while it provides more reliable and fine-grained regions in comparison with segmented objects.

SuperBIG creates and embeds superpixels of input and reference images in a unified bipartite graph framework. It obtains superpixels through two steps. The first one is to cluster pixels into superpixels based on distance measurement with dense correspondence, which is estimated by deep matching [49]. The relevant hierarchical features for measuring the distances between pixels include colors, intensity patterns, textures, *etc.* The second step is to employ an automatic bipartite partition in a unsupervised way to group pixels that are not covered by any superpixel in the first step. Here we elaborate on the related features.

We use the subscript (i,j) to index the pixel location of an image $\mathbf{I}$ and utilize superscript c and f to denote features of the input and reference images, respectively. $\mathbf{I}_{(i,j)}$ is defined as the intensity of a pixel at the location (i,j). We extract a set of features for the following two purposes: To measure the content similarity in the same domain/style (*e.g.* within an image) or to measure that cross domains/styles (*e.g.* in two styled images). Thus, the extracted features are classified into two categories: style-related (including patch intensity, color, gradient, absolute location) and style-independent (including texture, relative location, locality-constrained linear coding feature). All these extracted features are described below,

- Intensity vector of a patch:
$$\mathbf{M}_{(i,j)} = \left[\mathbf{I}_{(k,l)}\right]^T \Big|_{(k,l)\in\mathcal{N}_{(i,j)}},$$
where the set $\mathcal{N}_{(i,j)}$ contains locations of pixels (k,l) in a patch centered at the location (i,j).
- Color $\mathbf{C}_{(i,j)}$ at pixel (i,j), which is composed of,
$$\mathbf{C}_{(i,j)} = \left[\mathbf{I}_{\mathbf{R}(i,j)}, \mathbf{I}_{\mathbf{G}(i,j)}, \mathbf{I}_{\mathbf{B}(i,j)}\right]^T,$$
where $\mathbf{I_R}$, $\mathbf{I_G}$ and $\mathbf{I_B}$ are three channels of an image $\mathbf{I}$. They are related to the intensity of that pixel as follows,
$$\mathbf{I}_{(i,j)} = \sqrt{\mathbf{I}^2_{\mathbf{R}(i,j)} + \mathbf{I}^2_{\mathbf{G}(i,j)} + \mathbf{I}^2_{\mathbf{B}(i,j)}}.$$
- Gradient of a patch:
$$\mathbf{DV}_{(i,j)} = \left[\left\{\sqrt{\mathbf{dI}^2_{x(k,l)} + \mathbf{dI}^2_{y(k,l)}}\Big|(k,l)\in\mathcal{N}_{(i,j)}\right\}\right]^T,$$
where $\mathbf{dI}_x$ and $\mathbf{dI}_y$ denote the intensity variation of the original image along horizontal and vertical directions, respectively.
- Absolute location:
$$\mathbf{L}^a_{(i,j)} = \left[\frac{i}{h}, \frac{j}{w}\right]^T,$$
where h and w are the height and width of an image. It is defined as the normalized location in the original coordinates for the image.
- Texture feature $\mathbf{T}_{(i,j)}$ of a patch centered at pixel (i,j). The features of factorization-based texture segmentation [50] are extracted to segment different texture regions and locate their boundaries.
- Relative location, $\mathbf{L}^r_{(i,j)}$. SuperBIG regards the dense points as reliable locations and utilizes them to 'relocate' the pixels with the novel coordinates, which takes the locations of these matched points as the basis. It is defined as the representation coefficients of a pixel location, when taking locations of several nearest matched points within the image as the basis. Locations of five nearest matched points to pixel (i,j) are denoted as,
$$\boldsymbol{\tau} = \left\{[i_l, j_l]^T\,|_{l=1,2,\ldots,5}\right\}.$$
The current location (i,j) is represented by the multiplication of $\boldsymbol{\tau}$ and a representation coefficient $\boldsymbol{\alpha}$,
$$\boldsymbol{\tau}\boldsymbol{\alpha} = [i,j]^T.$$
Then, $\boldsymbol{\alpha}$ is solved by,
$$\boldsymbol{\alpha} = (\boldsymbol{\tau}^T\boldsymbol{\tau}I + n_{\boldsymbol{\alpha}})^{-1}(\boldsymbol{\tau}^T\,[i,j]^T),$$
where $n_{\boldsymbol{\alpha}}$ is the ridge parameter for $\boldsymbol{\alpha}$ to avoid singular solutions. To generate the relative location $\mathbf{L}^r_{(i,j)}$, we put the solved $\boldsymbol{\alpha}$ to $\mathbf{L}^r_{(i,j)}$ in the corresponding dimension that belongs to the matched point and zeros in other dimensions.
- Locality-constrained linear coding (LLC) feature, $\mathbf{S}_{(i,j)}$. Similar to the idea of calculating the relative location, we calculate the 'relative location' in the feature space, to generate a measurement of content similarity, independent on the style. Similarly, with the matched points provided by deep matching, we use features of these matched points as the basis (or the coordinates in the feature space) to calculate the representation coefficients, independent on the style. Assume the five nearest matched points at the location (i,j) are represented in the feature space,
$$\boldsymbol{\tau}_f = \left\{[\mathbf{M}_{i_l,j_l}, \mathbf{C}_{i_l,j_l}, \mathbf{I}_{i_l,j_l}, \mathbf{DV}_{i_l,j_l}]^T\,|_{l=1,2,\ldots,5}\right\}.$$
Then, a sparse coefficient $\boldsymbol{\beta}$ is calculated by solving,
$$\boldsymbol{\tau}_f\boldsymbol{\beta} = [\mathbf{M}_{i,j}, \mathbf{C}_{i,j}, \mathbf{I}_{i,j}, \mathbf{DV}_{i,j}]^T.$$
We then have,
$$\boldsymbol{\beta} = (\boldsymbol{\tau}_f^T\boldsymbol{\tau}_f + n_{\boldsymbol{\beta}}I)^{-1}(\boldsymbol{\tau}_f^T\,[\mathbf{M}_{i,j}, \mathbf{C}_{i,j}, \mathbf{I}_{i,j}, \mathbf{DV}_{i,j}]^T),$$
where $n_{\boldsymbol{\beta}}$ is the ridge parameter for $\boldsymbol{\beta}$ to avoid singular solutions. To generate the relative location $\mathbf{S}^r_{(i,j)}$, we put the solved $\boldsymbol{\beta}$ to $\mathbf{S}^r_{(i,j)}$ in the corresponding dimension that belongs to the matched point and zeros in other dimensions.
With the help of the above mentioned features of several nearest matched points $\mathbf{P}^c_{(i,j)}$ or $\mathbf{P}^f_{(i,j)}$, $\mathbf{S}^c_{(i,j)}$ and $\mathbf{S}^f_{(i,j)}$ are representation coefficients of the unmatched points (i,j) from the input and reference images.

Intuitively, these features are diverse in order to cover most information to build the content correspondence. As mentioned above, according to whether a feature is capable of measuring the content similarity cross styles, these features are classified into: style-related and style-independent. The former is mainly utilized to measure the similarity between input and reference images, while the latter is exploited to measure the similarity between two pixels in the same image.

Here we create superpixels around matched points and build a mapping based on the correspondences of these points.

Intuitively, coupled superpixels around paired matched points share the same style transformation. We use p and q to index two arbitrary pixels in the input and reference images, respectively. And let t index an arbitrary pixel in one of them. For each pair of matched point locations (i_p, j_p) and (i_q, j_q), the distance of one pixel (i_t, j_t) in the input image to the corresponding matched point in the reference image is calculated by style-dependent features as follows,

$$\begin{aligned} &\mathbf{D}^c(u_{(i_t,j_t)}, u_{(i_p,j_p)}) \\ = \; & -\frac{\left\|\mathbf{M}^c_{(i_t,j_t)} - \mathbf{M}^c_{(i_p,j_p)}\right\|_2^2}{\lambda_{\mathbf{M}}} - \frac{\left\|\mathbf{T}^c_{(i_t,j_t)} - \mathbf{T}^c_{(i_p,j_p)}\right\|_2^2}{\lambda_{\mathbf{T}}} \\ & -\frac{\left\|\mathbf{C}^c_{(i_t,j_t)} - \mathbf{C}^c_{(i_p,j_p)}\right\|_2^2}{\lambda_{\mathbf{C}}} - \frac{\left\|\mathbf{DV}^c_{(i_t,j_t)} - \mathbf{DV}^c_{(i_p,j_p)}\right\|_2^2}{\lambda_{\mathbf{DV}}} \\ & -\frac{\left\|\mathbf{L}^{a,c}_{(i_t,j_t)} - \mathbf{L}^{a,c}_{(i_p,j_p)}\right\|_2^2}{\lambda_{\mathbf{L^a}}}, \end{aligned} \tag{1}$$

where $\lambda_{(\cdot)}$ are weighting parameters to balance the effect of each term. The distance $\mathbf{D}^f(v_{(i_t,j_t)}, v_{(i_q,j_q)})$ in $\mathbf{I}^f$ can be computed similarly. Then, we create super-pixel clusters $\mathbf{F}^{c,m}_p$ and $\mathbf{F}^{r,m}_q$ containing all the pixels with a distance to p and q respectively less than a given threshold $\mathcal{T}_{cluster}$. After that, superpixels around the matched points are obtained. SuperBIG further deals with other unsettled pixels in a bipartite graph framework hereafter.

B. Pixel Bipartite Graph Partition

After obtaining the superpixel around matched points, SuperBIG constructs a pixel-level bipartite graph from the uncovered pixels that do not belong to any given superpixel. Afterward, a bipartite partition is followed to cluster those unsettled pixels into superpixels.

Let $\mathbf{f}^c_{(i,j)}$ and $\mathbf{f}^r_{(i,j)}$ represent the hierarchical features corresponding to the pixel located at (i, j) in the input and reference images. Because we aim to calculate the content closeness of pixels in two images with different styles, the hierarchical features consist of style-free features, such as locations, gradient, textures, defined as follows,

$$\mathbf{f}^c_{(i,j)} = \left[\mathbf{S}^c_{(i,j)}, \mathbf{T}^c_{(i,j)}, \mathbf{L}^{a,c}_{(i,j)}, \mathbf{L}^{r,c}_{(i,j)}\right]. \tag{2}$$

So does $\mathbf{f}^r_{(i,j)}$.

Based on the hierarchical features to calculate the affinities between nodes, SuperBIG constructs the pixel bipartite graph. Let $u_{(i,j)}$ and $v_{(i,j)}$ denote the node corresponding to the pixel in the location (i, j) of the input and reference image, respectively. Here (i, j) only represents the location of unsettled pixels. There is an edge connection between corresponding nodes in the bipartite graph, only when the nearest dense points of their corresponding pixels are largely matched. Then, the pixel corresponds to the node in the graph, and edge weights (affinities) are calculated based on hierarchical features $\mathbf{f}^c_{(i_p,j_p)}$ and $\mathbf{f}^f_{(i_q,j_q)}$ adjusted by weighting parameters $\lambda_{(\cdot)}$ for each kind of features as follows,

$$\begin{aligned} &\mathbf{E}(u_{(i_p,j_p)}, v_{(i_q,j_q)}) \\ &= \exp\left\{ -\frac{\left\|\mathbf{S}^c_{(i_p,j_p)} - \mathbf{S}^f_{(i_q,j_q)}\right\|_2^2}{\lambda_{\mathbf{S}}} - \frac{\left\|\mathbf{T}^c_{(i_p,j_p)} - \mathbf{T}^f_{(i_q,j_q)}\right\|_2^2}{\lambda_{\mathbf{T}}} \right. \\ &\left. - \frac{\left\|\mathbf{L}^{a,c}_{(i_p,j_p)} - \mathbf{L}^{a,f}_{(i_q,j_q)}\right\|_2^2}{\lambda_{\mathbf{L^a}}} - \frac{\left\|\mathbf{L}^{r,c}_{(i_p,j_p)} - \mathbf{L}^{r,f}_{(i_q,j_q)}\right\|_2^2}{\lambda_{\mathbf{L^r}}} \right\}. \end{aligned} \tag{3}$$

Then, a weighted bipartite graph is constructed between two nodes (u, v), corresponding to the pixels of images that are exactly paired matched points in the dense correspondence. Their edge weights (affinities) $\mathbf{E}(u, v)$ correspond to the similarities, which are independent of the style.

When performing the graph partition, a natural choice is spectral clustering. It is exploited to capture the cluster structure of a graph by clustering the spectrum of the Laplacian matrix. $\mathbf{D}$ is defined as the degree matrix. It is formulated as a generalized eigen-problem,

$$\mathbf{Jg} = \lambda \mathbf{Dg}, \tag{4}$$

where λ is the eigenvalue to be optimized. And $\mathbf{J} = \mathbf{D} - \mathbf{\Omega}$ is the Laplacian matrix and $\mathbf{D} = diag(\mathbf{\Omega 1})$ is the degree matrix. $\mathbf{1}$ is a unit vector and $\mathbf{\Omega}$ denotes the affinity (adjacent) matrix of the graph, that contains the affinity $\mathbf{E}(u, v)$ of every paired nodes (u, v) in the graph. For clustering, the Laplacian matrix is approximated by a block-diagonal matrix including k eigenvalues block-diagonal matrix. The Laplacian matrix can be also defined as the normalized Laplacian $\mathbf{J}_N = \mathbf{D}^{-1/2}\mathbf{J}\mathbf{D}^{-1/2}$ or generalized Laplacian $\mathbf{J}_G = \mathbf{D}^{-1}\mathbf{J}$.

It can be solved with the Lanczos method [51] on the normalized affinity matrix $\tilde{\mathbf{\Omega}} = \mathbf{D}^{-1/2}\mathbf{\Omega}\mathbf{D}^{1/2}$ or partial SVD [52] on normalized across-affinity matrix. Adopting the latter solution in our method, the bottom k eigenvectors of (4) are obtained by the top k left and right singular vectors of the normalized across-affinity matrix,

$$\tilde{\mathbf{\Omega}}_a = \mathbf{D}_X^{-1/2}\mathbf{\Omega}\mathbf{D}_Y^{-1/2}, \tag{5}$$

where $\mathbf{D}_X = diag(\mathbf{\Omega})\mathbf{1}$ and $\mathbf{D}_Y = diag(\mathbf{\Omega})^T\mathbf{1}$ denote the degree matrix of $\mathbf{X}$ and $\mathbf{Y}$, respectively. Then, we obtain k superpixel clusters $\mathbf{F}^{c,u}_p$ and $\mathbf{F}^{r,u}_q$ and get a set of coupled superpixel clusters $\mathbf{F}^c = [\mathbf{F}^{c,m}, \mathbf{F}^{c,u}]$ and $\mathbf{F}^r = [\mathbf{F}^{r,m}, \mathbf{F}^{r,u}]$.

C. Superpixel Bipartite Graph Matching

In the above step, SuperBIG estimates the superpixels for the pixels that are not covered by superpixels of matched points. In this process, superpixels of matched points and their covered pixels are totally ignored in the constructed pixel-level bipartite graph. It may lead to inaccurate matchings when some superpixels of matched pixels in the input image in fact correspond to the superpixels of unmatched pixels in the reference image.

Thus, SuperBIG constructs a superpixel bipartite graph and performs a graph matching on it. The nodes of the new graph represent superpixels of $\mathbf{F}^c$ and $\mathbf{F}^r$. There is an edge connection between corresponding nodes, only when their hierarchical features are close enough in the feature space. Considering that the pixels in a superpixel share similar features, for similarity, hierarchical features of a superpixel are defined as the mean vector of hierarchical features of pixels within it. And the affinities between superpixel bipartite graph are calculated based on the superpixel hierarchical feature, in the same way as (3). Then, SuperBIG solves the bipartite graph matching by the Hungarian algorithm [53], obtaining final superpixel correspondences $\mathbf{F}^c_f$ and $\mathbf{F}^r_f$.

D. De-Correlated Style Transfer

After we obtain a reliable superpixel correspondence, the style transfer based on such a correspondence is built. Color and contrast transfer usually changes the dominant color and contrast distribution, and maps to desirable color and contrast casts. A slightly more general approach is to fit the color statistic of the input image into that of the reference one. Global methods based on the color statistic cannot handle some tough cases, such as the image containing complex details and diverse colors. Based on the SuperBIG framework, the styles of an image could be transferred locally at the granularity of superpixel.

SuperBIG transfers colors by manipulating the statistic in the $l\alpha\beta$-CIE space, a de-correlated color space, as our local mapping method. Here we define $[\mathbf{I}_L, \mathbf{I}_M, \mathbf{I}_S]^T = \mathcal{F}[\mathbf{I}_R, \mathbf{I}_G, \mathbf{I}_B]^T$, where $\mathcal{F}$ is a predefined transformation matrix and $\mathbf{I}_R, \mathbf{I}_G, \mathbf{I}_B$ are three channels of a RGB image. Then, we convert $[\mathbf{I}_L, \mathbf{I}_M, \mathbf{I}_S]^T$ to the logarithmic space,

$$\mathbf{I_L} = log\mathbf{I}_L,\ \mathbf{I_M} = log\mathbf{I}_M,\ \mathbf{I_S} = log\mathbf{I}_S,$$

$$\begin{bmatrix} \mathbf{I}_l \\ \mathbf{I}_\alpha \\ \mathbf{I}_\beta \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -2 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{I_L} \\ \mathbf{I_M} \\ \mathbf{I_S} \end{bmatrix}.$$

This decorrelation makes three color channels independent. SuperBIG then adjusts the color statistic in such space by matching mean and variance as follows,

$$\begin{aligned} \mathbf{I}^\star_l &= \mathbf{I}_l - \langle \mathbf{I}_l \rangle,\ \mathbf{I}^\star_\alpha = \mathbf{I}_\alpha - \langle \mathbf{I}_\alpha \rangle,\ \mathbf{I}^\star_\beta = \mathbf{I}_\beta - \langle \mathbf{I}_\beta \rangle, \\ \mathbf{I}'_l &= \frac{\sigma^l_r}{\sigma^l_c}\mathbf{I}^\star_l,\ \mathbf{I}'_\alpha = \frac{\sigma^\alpha_r}{\sigma^\alpha_c}\mathbf{I}^\star_\alpha,\ \mathbf{I}'_\beta = \frac{\sigma^\beta_r}{\sigma^\beta_c}\mathbf{I}^\star_\beta, \end{aligned} \tag{6}$$

where $\langle\cdot\rangle$ is the operator to calculate the mean and σ is the variance of the image for a given channel. With the de-correlated style transfer for local regions, SuperBIG transfers styles between each pair of estimated corresponding super-pixel pairs in $\mathbf{F}^c_f$ and $\mathbf{F}^r_f$. After that, we finally smooth the transferred results by the guided image filter [54]. It helps remove the boundary effect between superpixels, as well as detailed artifacts caused by inaccurate super-pixel mappings when the hierarchical features of super-pixels fail to describe their relationship.

IV. Experimental Results

A. Experimental Setting

We compare the proposed method (SuperBIG) with the following eight state-of-the-art style/color transfer methods: L$\alpha\beta$ decorrelated color space (L$\alpha\beta$) [1], color "mood" transfer (MoodTrans) [55], multi-scale harmonization (Harmonization) [7], landmark sparse color representation (Landmark) [34], neural algorithm of artistic style (NeutralArt) [13], superpixels matching (SuperMatch) [16], image morphing + SITF flow (Image Morphing) [41] and data-driven hallucination (Data-driven) [42]. To make a fair comparison in our case, for Image Morphing, the initial matched points are provided by deep matching and no foreground and background masks are used. For data-driven hallucination, the coupled references are replaced with our input and referenced images, without the aid of additional video resources. Results of these methods are generated by the published codes kindly provided by the authors. When compared to the colorization methods, SuperBIG first turns the input image into greyscale one, then colorizes the generated greyscale image. We use two sets of parameters for two tasks respectively in our experiments in Table I.

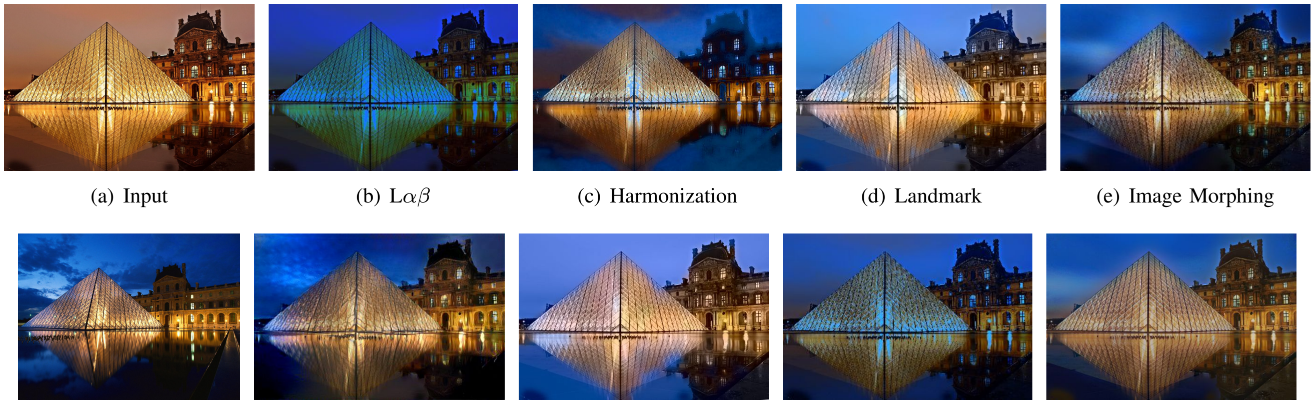
TABLE I
THE PARAMETER SETTINGS FOR TWO TASKS RESPECTIVELY IN OUR EXPERIMENTS.

Parameters	$\lambda_\mathbf{M}$	$\lambda_\mathbf{T}$	$\lambda_\mathbf{C}$	$\lambda_\mathbf{DV}$	$\lambda_\mathbf{S}$
Similar scenes	0.1	0.001	0.0001	10^{-6}	0.1
Different scenes	0.1	0.01	0.01	10^{-6}	0.1
Parameters	$\lambda_{\mathbf{L^a}}$	$\lambda_{\mathbf{L^r}}$	$n_{\boldsymbol{\alpha}}$	$n_{\boldsymbol{\beta}}$	-
Similar scenes	0.01	0.01	1000	10^6	-
Different scenes	0.001	0.001	10	10^6	-

These parameters are initialized as $\lambda_\mathbf{M} = 0.01, \lambda_\mathbf{T} = 1, \lambda_\mathbf{C} = 0.01, \lambda_\mathbf{DV} = 0.01, \lambda_\mathbf{S} = 1, \lambda_{\mathbf{L^a}} = \lambda_{\mathbf{L^r}} = 1, n_{\boldsymbol{\alpha}} = 10^4$ and $n_{\boldsymbol{\beta}} = 10^4$. Then, we increase/decrease parameters of some terms gradually based on the performance of each task. Note that, we do not tune parameters for each paired inputs.

B. Comparison with State-of-the-Art and Various Styles

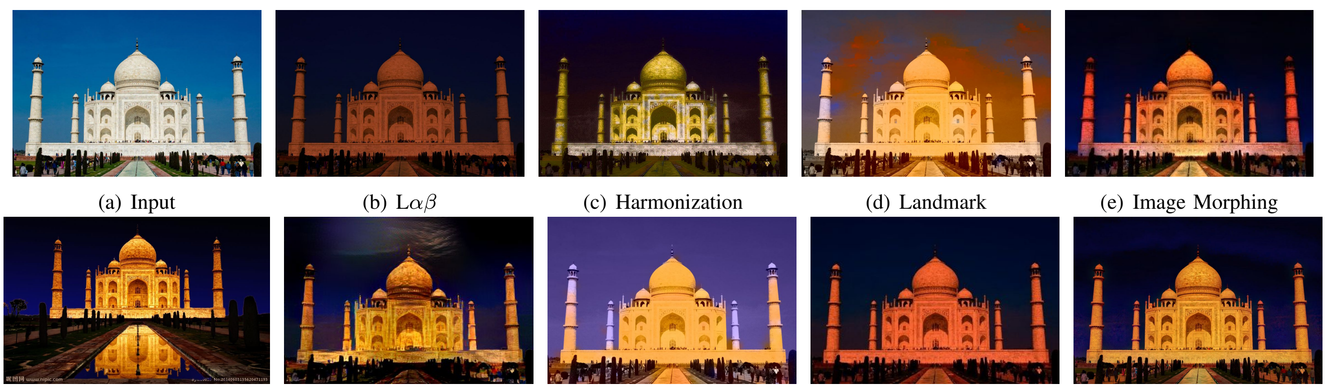
The comparison results of SuperBIG and other state-of-the-art methods for three input images are presented in Figs. 2-4. Please enlarge and view these figures on the screen for better comparison. The subjective quality of these results demonstrates the superiority of the proposed SuperBIG. L$\alpha\beta$ and Harmonization totally fail to transfer the color, because of wrong dominant color prediction in Figs. 2(b) and 3(b) as well as heavily blurred or extremely rough sky regions in Figs. 2(c)-4(c), respectively. Landmark, NeutralArt and SuperMatch suffer from wrong local style predictions, *e.g.* blue color near the edges and corners of the pyramid in Figs. 2(d)(f)(g) and the color artifacts on the top of the towers of Taj Mahal in Figs. 4(d)(f)(g). For image morphing [41], lights and contrasts in regions are transferred well, however, it suffers at boundaries between regions, where wrong color transfers contaminate the transfered results. For data-driven hallucination [42], without the guidance of additional coupled video sequences, it is easy for that method to degenerate to a global transfer. Thanks to informative hierarchical features



(a) Input (b) L$\alpha\beta$ (c) Harmonization (d) Landmark (e) Image Morphing

(f) Reference (g) NeutralArt (h) SuperMatch (i) Data-driven (j) SuperBig

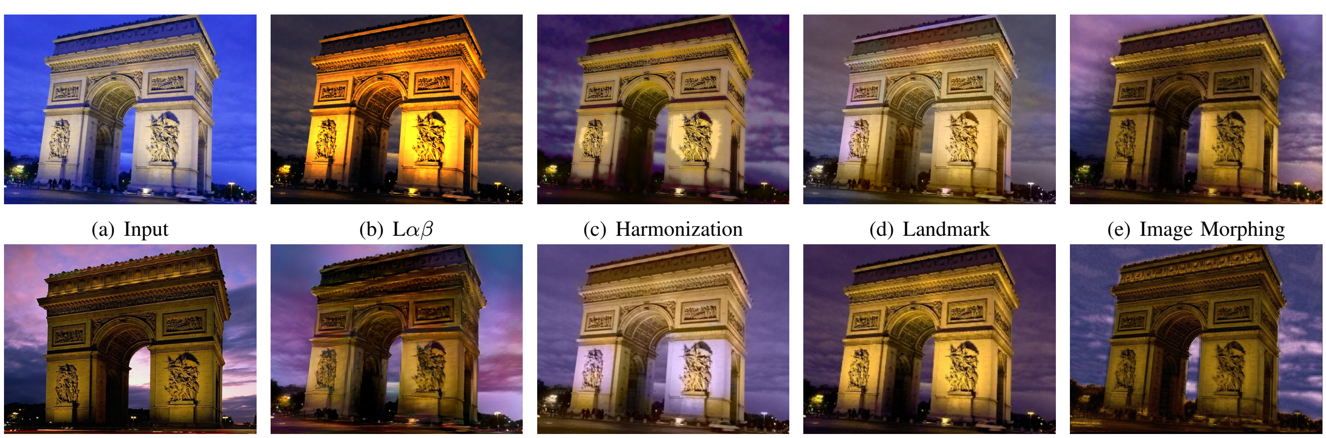
Fig. 2. Visual comparisons of style transfer among different algorithms.



(a) Input (b) L$\alpha\beta$ (c) Harmonization (d) Landmark (e) Image Morphing

(f) Reference (g) NeutralArt (h) SuperMatch (i) Data-driven (j) SuperBig

Fig. 3. Visual comparisons of style transfer from (a) to (e) among different algorithms.



(a) Input (b) L$\alpha\beta$ (c) Harmonization (d) Landmark (e) Image Morphing

(f) Reference (g) NeutralArt (h) SuperMatch (i) Data-driven (j) SuperBig

Fig. 4. Visual comparisons of style transfer from (a) to (e) among different algorithms.

and effective superpixel bipartite framework for modeling in the global and local correspondences, SuperBIG transfers the proper styles for the local regions in the generated results as shown in Figs. 2(j)-4(j).

The subjective results of SuperBIG to transfer different styles are showed in Fig. 5. From the results, we observe that SuperBIG generates the results containing clear and natural content while successfully changing their styles based on the reference images, leading to similar spatial distribution of color and contrast. It is worth noting that, even for the night image as shown in the right-bottom of Fig. 5(b), where background light is dim, SuperBIG can still achieve the transformation successfully and generate naturally looking results.

C. User Study in Subjective Evaluation

To compare different stylization results from an observer's perspective, we employ the paired comparisons approach, where the participants are shown two stylized images at a time, side by side, and are asked to simply choose the preferred one by considering both visual quality and similar style to the exemplar. We have a total of 90 participants, including both domain experts and generally knowledgeable individuals, each given 105 pairwise comparisons over a set of five images with seven different style transfer methods. Fig. 6 illustrates the seven methods, ranked by the number of votes received. It can be seen that the proposed SuperBIG outperforms other methods in four out of the five cases, and achieves overall superior performance. Even in the exceptional case with the test image *Arch*, it still shows comparable performance with the first ranked method. Besides the voting statistic, we also show the stability analysis, which is calculated by the *rank product* [56]. Table II shows the results of the rank product $\psi(O) = (\prod_i r_{O,i})^{1/b}$, where $r_{O,i}$ is the specific ranking for method O and image i $(i = 1 \dots b)$. Compared with others, SuperBIG produces the best consistency among different test cases to achieve the best visual quality.

D. Ablation Analysis

To further explore the functionality of each step of SuperBIG, we perform the ablation analysis of each step in the flowchart as shown in Fig. 7. We find that deep matching provides a large amount of matched points. It can be observed from Figs. 7(b)(g) that most of them are visually correct. Taking a given portion of matched points (70% with highest confidence scores) and calculating the hierarchical features, SuperBIG obtains superpixels around matched points as shown in Figs. 7(c)(h). Afterwards, uncovered pixels are handled in a pixel-level bipartite graph to generate other superpixels in Figs. 7(d)(i). According to the correspondence obtained so far, we generate the style transfer result of Fig. 7(e). It can be seen that, because the matching from the previous steps does not consider the global information, it generates only the locally consistent result. There are some visually unpleasant details. First, there are some inaccurate color transfer results in the right- bottom part of the image. Second, the sky in Fig. 7(e) presents abundant textures, different from that in both the input and reference images. Thus, SuperBIG reconsiders the matching between all superpixels of the two images. Due to the feature refined from pixels to superpixels and global optimization, SuperBIG generates a well-constructed result in Fig. 7(j).

E. Visual Comparison for Local Transfer

To evaluate the effectiveness of our bipartite graph framework for local transfer, we compare it with the state-of-the-art local transfer method - image morphing with SIFT-flow (Image Morphing) [41], as shown in Fig. 8. From the results, it is clearly shown that our method achieves superior visual quality. The proposed hierarchical feature covers many factors including color, gradient, textures, *etc.*, and the bipartite partition and matching utilize the context information across the whole image. Based on them, with moderately accurate geometric correspondences, our method still achieves promising results. Comparatively, in the general case, there is no reliable matched points, and Image Morphing is not robust to inaccurate matchings and suffers from the inaccurate transfer at the boundaries between regions.

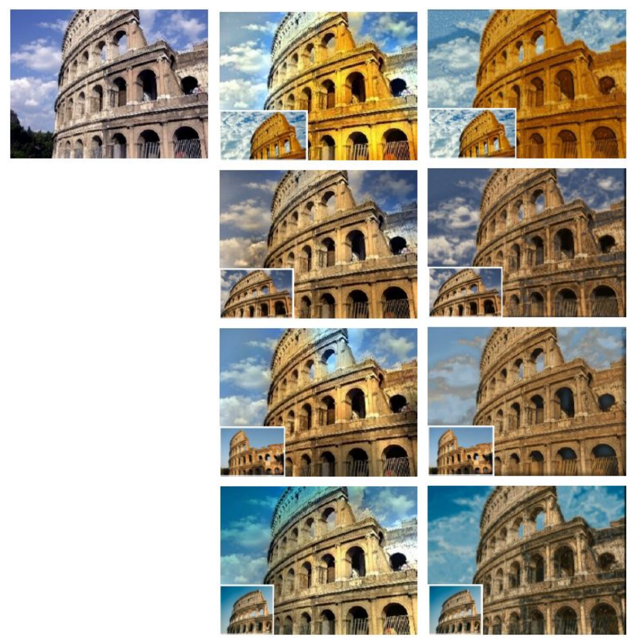
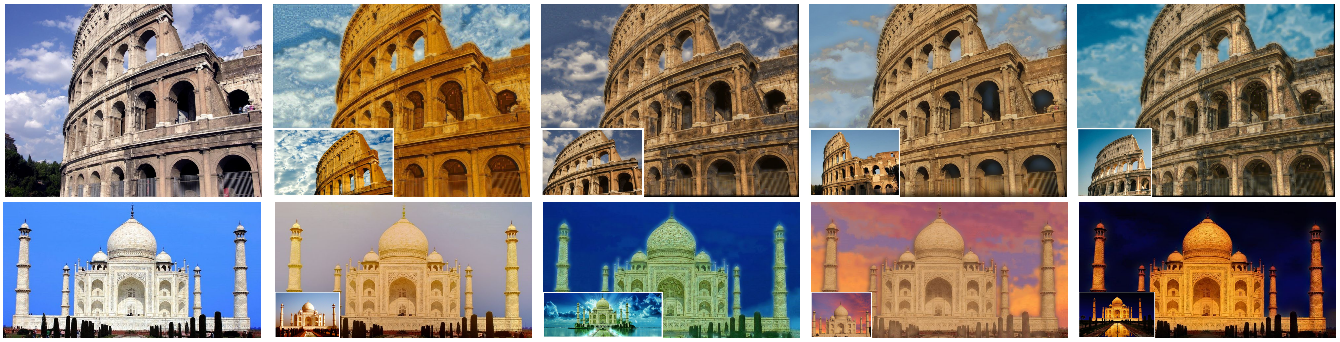


Fig. 8. Visual comparisons for local transfer with image morphing and SIFT flow (Image Morphing) [41]. In each group, from left to right, each column is the input image, the result of Image Morphing and that of SuperBIG, respectively. The inserts show the reference images.

F. Visual Results in Different Scenes

Constrained by the local and global consistency, our two-step bipartite graph framework is capable of removing false initial matchings, creating reliable matchings robustly and generating natural-looking stylized results. Thus, it is capable to transfer the style in partial occluded image or highly affined image in some extent. The results of our method with scene changes are illustrated in Fig. 9. It is observed that, with our hierarchical features to encode low to high level context information and bipartite graph framework that effectively handles initial wrong matchings, our approach achieves rather good visual quality in scene change cases.



(a) **Input** (b) **Output**: Styles transferred photos from the examples. The inserts show the examples.

Fig. 5. Visual comparisons of SuperBIG style transfer for different reference images.

TABLE II
COMPARISON OF THE RANK PRODUCT OF SEVEN METHODS.

Method	CIE-L$\alpha\beta$	Harmonization	Landmark	MoodTrans	NeutralArt	SuperMatch	SuperBIG
Rank ψ	4.04	5.07	4.22	6.35	2.83	2.83	**1.15**

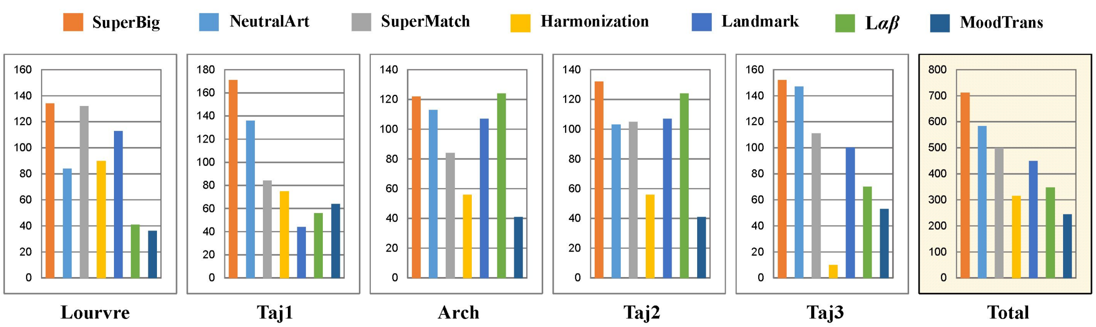


Fig. 6. The number of votes per testing image and the total ranking of seven methods.

V. FURTHER ANALYSIS AND DISCUSSIONS

In this section, more analysis about our method and the related results are presented. Besides, the limitations of our method and several potential future directions are briefly discussed.

A. Computational Complexity and Acceleration

We report the time cost of our SuperBIG and compare its efficiency with other methods. The compared methods are from the public available codes provided by the authors. The Neutral Art is implemented in Torch and we run it with GPU for testing. Our proposed method and other compared methods are implemented in MATLAB. We evaluate the running time of all the algorithms with the following machine configuration: Intel i5-3230M 2.60GHz and 12 GB memory. Table III presents the running time of one transfer with an input (800×600) and a reference (800×761) image for all comparison methods.

SuperBIG is the unaccelerated version of the proposed method. SuperBIG+PM is the version with Matlab built-in parallelization and a zooming acceleration. The zooming acceleration is to down-sample the input and reference images, and then transfer the style at a small scale. Then, the transferred result is up-sampled to its original size and finally a detail enhancement in its luminance channel is utilized. From the table, it is observed that, time cost of our proposed method without any acceleration remains similar to SuperMatch and Neutral Art in magnitudes. The parallelization and zooming acceleration significantly improve the efficiency of our method, achieving the competitive time efficiency to Harmonization and Landmark.

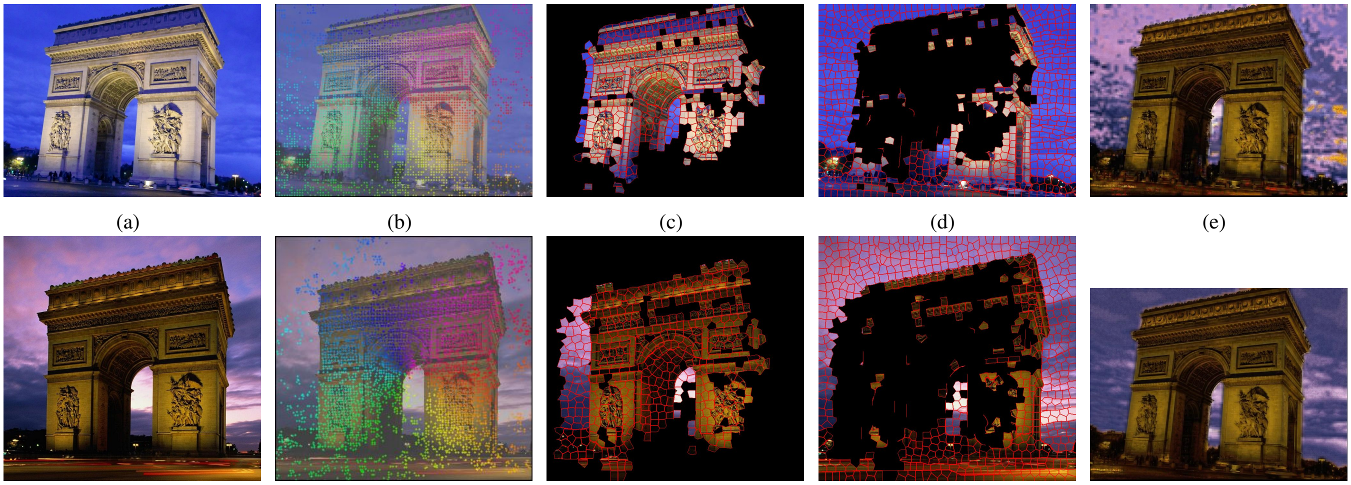
With these acceleration techniques, our SuperBIG is reaching the application with minute-level running time. The ways to further accelerate our method to facilitate a real-time application, such as GPU implementation [57], [58] or precomputed partition and matching of certain sub-graphs with specific structures, are worthy of our future exploration.

TABLE III
RUNNING TIME (S) OF ALL METHODS.

Methods	Harmonization	Landmark	Neutral
Running time (s)	98.3438	123.9686	422.0246
Methods	SuperMatch	SuperBIG	SuperBIG+PM
Running time (s)	2056.6673	754.2880	183.6594

B. Visual Results for Video Stylization

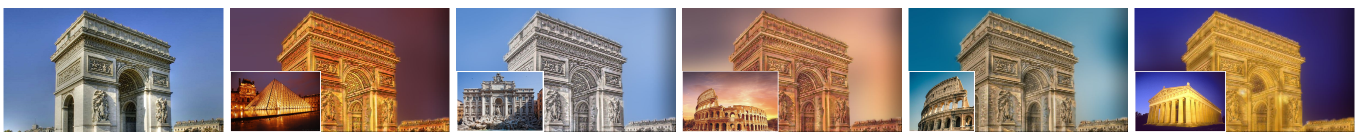
To evaluate the generality of our SuperBIG, we apply our method for video stylization with a simple smooth between the transfer parameters of nearest superpixels among adjacent frames. Our method achieves rather impressive results, even



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

(f) (g) (h) (i) (j)

Fig. 7. The ablation analysis for SuperBIG. (a) The input image. (b) Dense correspondence in (a). (c) Superpixels for matched points in (a). (c) Superpixels for other pixels in (a). (e) The transfer results with the superpixel correspondece generated from the pixel-level bipartite graph partition. (f) The reference image. (g) Dense correspondence in (f). (h) Superpixels for matched points in (f). (i) Superpixels for other pixels in (f). (j) The transfer results with the superpixel correspondece generated from the superpixel-level bipartite graph matching.



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

Fig. 9. The results of the proposed SuperBIG to transfer the style with scene changes (Input and reference images are totally different scenes). (a) Input. (b)-(f) Outputs: Style transferred photos from the examples in totally different scenes. The inserts show the examples.

for a video with large view changes. Some shots are shown in Fig. 10.

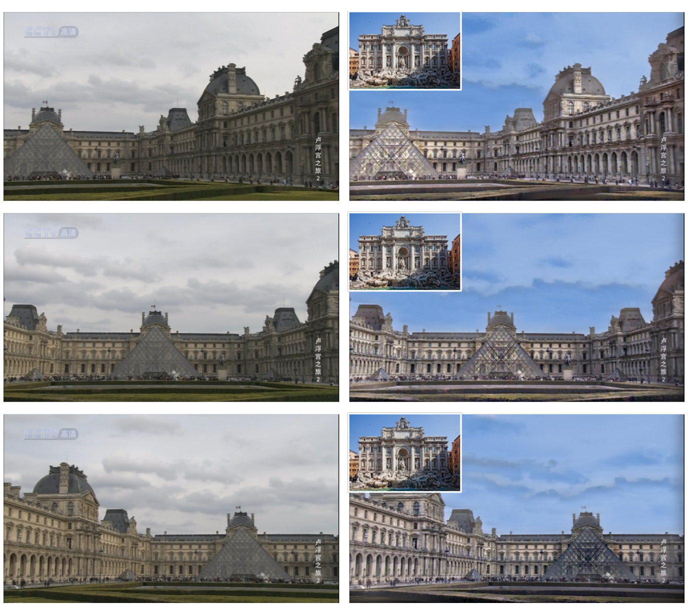


Fig. 10. Video stylization results with the camera shift. From left to right, each collumn is the input frames and stylized frames, respectively. From top to bottom, each row is the $1st$, $101st$ and $201st$ frame, respectively. The inserts show the reference image.

C. Hierarchical Features Analysis

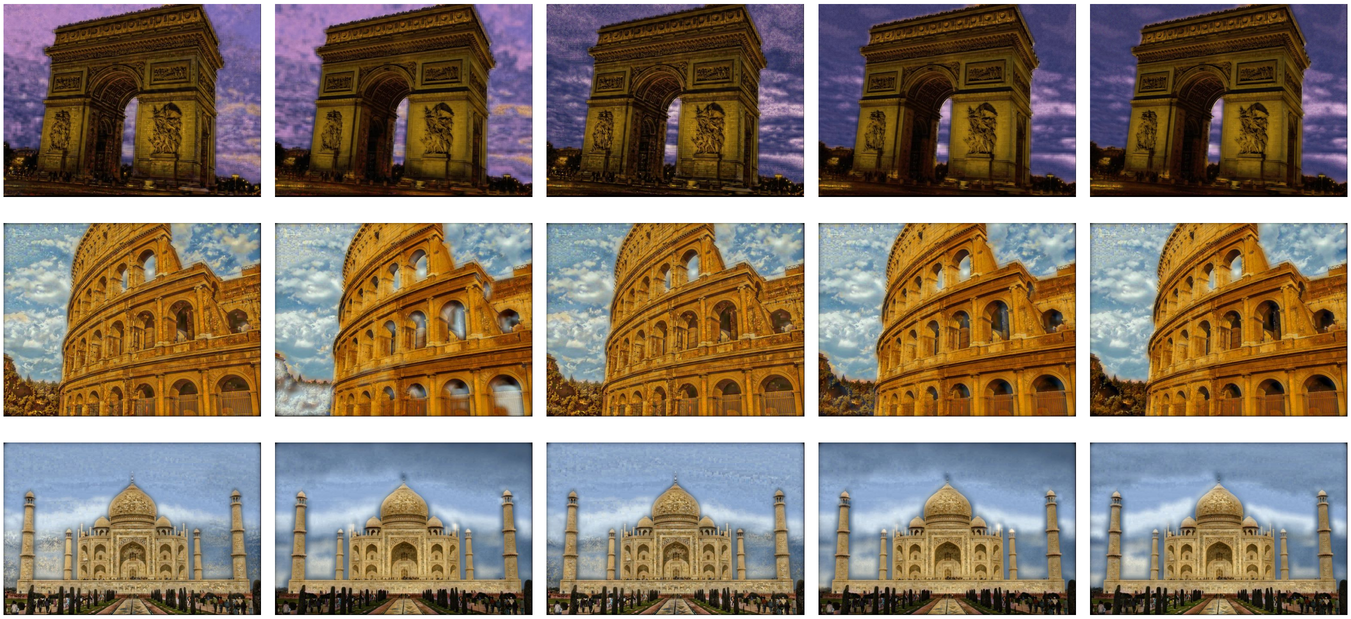
We also explore the effectiveness of each feature in the hierarchical features. Our features are selected by three steps. Frist, we analyze and follow the observations in previous methods. Thus, the patch intensity, color, gradient and locations are selected to represent low-level features, the texture is utilized to represent mid-level features. Second, we develop a robust combination by concatenating these features and encoding them as the high level feature via the locality-constrained linear coding (LLC). Finally, we observe and compare transfer performance and adjust the combination empirically.

Here, we only focus on the functionality of primitive features: color, distance (absolute and relative), texture, patch intensity vector, gradient. Fig. 11 shows the results generated by SuperBIG with the compositions of these five features. From the results, it could be seen that the composition of color and distance, or patch intensity vector alone leads to the result containing many falsely transferred regions. Adding the texture feature removes many false regions by texture consistency. However, the quality of the sky is limited. The patch intensity vector puts the local constraint on the transfer and generates naturally looking result. The gradient feature generates a more smooth result with a higher visual quality.

The endeavor to evaluate the quality of the stylization [59], [60] has caught our attentions and inspired us to explore selecting a beneficial combination automatically from a large number of candidates to form the hierarchal features, by observing the performance on an evaluation set with an appropriate metric to measure the transfer quality.

D. Comparision with Deep Learning-based Approaches

By encoding from low to high contexts, deep features locate and connect each object in the input and reference images.



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

Fig. 11. The validation of the hierarchical features in SuperBIG. (a) Color + Distance. (b) Patch intensity vector. (c) Color + Distance + Texture. (d) Color + Distance + Texture + Patch intensity vector. (e) Color + Distance + Texture + Patch intensity vector + Gradient.

However, their major weakness is its incapacity to restore good local details. As shown in Fig. 13, NeuralDoodle [61] and NeutralArt [13] tend to generate texture-like artifacts in the transferred results. Comparatively, benefiting from our hierarchical features that also encode low- to high-level contexts and our flexible and robust bipartite graph framework, our method generates the transferred results with both global and local consistency.

To further boost the effectiveness of our hierarchical features by combing our current version and deep ones, which are indeed proven very effective to encode object-level information in previous works, is worthy of the future exploration.

E. Comparison with Global Color Transfer Methods

To demonstrate the importance of modeling local transfer, we compare to global color transfer methods on testing cases including diversified color distributions as shown in Fig. 12(a). In these cases, the global transfer mapping, including IDT [4] (Fig. 12(b)) and Monge-Kantorovitch Linear colour mapping (MKL) [62] (Fig. 12(c)), is not enough to describe the complex mappings and transfers wrong colors between objects. Comparatively, SuperBIG (Fig. 12(d)) generates locally similar results in color and textures, such as the leaf in the first row, the cloud and ground in the second row, and the sky and grass in the third row.

F. Failure Cases and Potential Directions

Our style transfer is sensitive to the initial matched points. As shown in Fig. 14, due to a large amount of mismatched points between the input and reference images from the building to the sky, SuperBIG colors the sky with golden color. Thus, our future efforts will make the algorithm more

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

Fig. 12. Visual comparisons of style transfer with global color transfer methods. (a) Input and reference images. (b) IDT [4]. (c) MKL [62]. (d) SuperBIG.

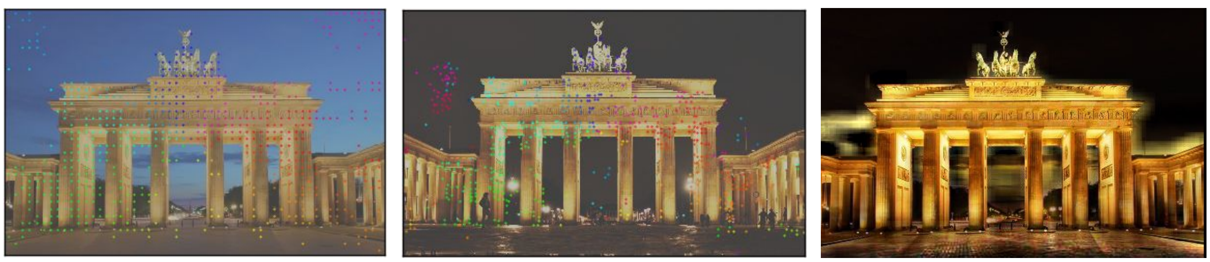
independent on the initial mappings and further robust to the wrong initialization. Besides, there are some works [41], [42] showing the superiority of utilizing SIFT flow to provide the initialized matchings and using image morphing to gradually map the reference style to the input one. Thus, it is also interesting to revisit SuperBIG to embed the design methodology of SIFT flow and image morphing, or to combine them with SuperBIG to construct a more general and robust style transfer method.

VI. Conclusion

In this paper, we first introduce the concept of image stylistic brush and accordingly design an exemplar-based photo stylization method, SuperBIG, powered by a two-step bipartite graph algorithm. Specifically, a bipartite graph is constructed by considering dense correspondence and hierarchical features to partition pixels of the input and reference images into

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

Fig. 13. Visual comparisons with deep learning-based methods. (a) Input. (b) NeuralDoodle [61]. (c) NeutralArt [13]. (d) SuperBIG. (e) Reference.



(a) Input image with matched points (b) Reference image with matched points (c) Transferred results

Fig. 14. A failure case of SuperBIG for style transfer. Due to a large amount of mismatched points between the input and reference images from the building to the sky, SuperBIG colors the sky with golden color.

superpixels first. Then, we generate a superpixel-level bipartite graph, which produces correspondences of the superpixels by bipartite matching. The correspondence is then used to guide the style transformation in a decorrelated color space. Extensive experimental results demonstrate that the proposed SuperBIG method achieves superior visual quality compared to state-of-the-art methods while providing style consistent with the reference image.

REFERENCES

[1] E. Reinhard, M. Ashikhmin, B. Gooch, and P. Shirley, "Color transfer between images," *IEEE Computer graphics and applications*, no. 5, pp. 34–41, 2001.

[2] B. Wang, Y. Yu, T.-T. Wong, C. Chen, and Y.-Q. Xu, "Data-driven image color theme enhancement," in *ACM Trans. Graphics*, vol. 29, no. 6, 2010, p. 146.

[3] T. Pouli and E. Reinhard, "Progressive histogram reshaping for creative color transfer and tone reproduction," in *Proc. Int'l symposium on Non-photorealistic animation and rendering*, 2010, pp. 81–90.

[4] F. Pitié, A. C. Kokaram, and R. Dahyot, "Automated colour grading using colour distribution transfer," *Computer Vision and Image Understanding*, vol. 107, no. 1, pp. 123–137, 2007.

[5] S. Bae, S. Paris, and F. Durand, "Two-scale tone management for photographic look," in *ACM Trans. Graphics*, vol. 25, no. 3, 2006, pp. 637–645.

[6] Y. Li, L. Sharan, and E. H. Adelson, "Compressing and companding high dynamic range images with subband architectures," *ACM Trans. Graphics*, vol. 24, no. 3, pp. 836–844, 2005.

[7] K. Sunkavalli, M. K. Johnson, W. Matusik, and H. Pfister, "Multi-scale image harmonization," in *ACM Trans. Graphics*, vol. 29, no. 4, 2010, p. 125.

[8] A. Hertzmann, "A survey of stroke-based rendering," *IEEE Computer Graphics and Applications*, no. 4, pp. 70–81, 2003.

[9] X. Wang and X. Tang, "Face photo-sketch synthesis and recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 31, no. 11, pp. 1955–1967, 2009.

[10] C. J. Curtis, S. E. Anderson, J. E. Seims, K. W. Fleischer, and D. H. Salesin, "Computer-generated watercolor," in *Proc. Conf. on Computer graphics and interactive techniques*, 1997, pp. 421–430.

[11] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin, "Image analogies," in *Proc. Int'l Conf. on Computer graphics and interactive techniques*, 2001, pp. 327–340.

[12] W. Zhang, C. Cao, S. Chen, J. Liu, and X. Tang, "Style transfer via image component analysis," *IEEE Trans. on Multimedia*, vol. 15, no. 7, pp. 1594–1601, 2013.

[13] L. A. Gatys, A. S. Ecker, and M. Bethge, "A neural algorithm of artistic style," *Arxiv:*, vol. abs/1508.06576, 2015. [Online]. Available: http://arxiv.org/abs/1508.06576

[14] W. Cheng, R. Jiang, and C. W. Chen, "Color photo makeover via crowd sourcing and recoloring," in *Proc. ACM Int'l Conf. Multimedia*, 2015, pp. 943–946.

[15] A. Y.-S. Chia, S. Zhuo, R. K. Gupta, Y.-W. Tai, S.-Y. Cho, P. Tan, and S. Lin, "Semantic colorization with internet images," in *ACM Trans. Graphics*, vol. 30, no. 6, 2011, p. 156.

[16] R. K. Gupta, A. Y.-S. Chia, D. Rajan, E. S. Ng, and H. Zhiyong, "Image colorization using similar images," 2012, pp. 369–378.

[17] T. Pouli and E. Reinhard, "Progressive color transfer for images of arbitrary dynamic range," *Computers & Graphics*, vol. 35, no. 1, pp. 67–80, 2011.

[18] V. Bychkovsky, S. Paris, E. Chan, and F. Durand, "Learning photographic global tonal adjustment with a database of input / output image pairs," in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, June 2011.

[19] A. Adams, *The Negative: Exposure and Development Basic Photo 2*. Morgan and Lester, Eds., 1948.

[20] ——, *The Print: Contact Printing and Enlarging*. Morgan and Lester, Eds., 1950.

[21] K. Eisman and W. Palmer, *Adobe Photoshop Restoration and Retouching*. N. R. Press, Ed., 2006.
[22] D. Margulis, *Photoshop LAB Color: The Canyon Conundrum and Other Adventures in the Most Powerful Colorspace*. P. Press, Ed., 2005.
[23] G. Winkenbach and D. H. Salesin, "Computer-generated pen-and-ink illustration," in *Proc. Conf. on Computer graphics and interactive techniques*, 1994, pp. 91–100.
[24] J. E. Kyprianidis, J. Collomosse, T. Wang, and T. Isenberg, "State of the art: A taxonomy of artistic stylization techniques for images and video," *IEEE Trans. on Visualization and Computer Graphics*, vol. 19, no. 5, pp. 866–885, 2013.
[25] W. Li, M. Agrawala, B. Curless, and D. Salesin, "Automated generation of interactive 3D exploded view diagrams," in *ACM Trans. Graphics*, vol. 27, no. 3, 2008, p. 101.
[26] A. Hertzmann, "Paint by relaxation," in *Proc. Int'l Conf. Computer Graphics*, 2001, pp. 47–54.
[27] L. Lin, K. Zeng, H. Lv, Y. Wang, Y. Xu, and S.-C. Zhu, "Painterly animation using video semantics and feature correspondence," in *Proc. Int'l symposium on Non-photorealistic animation and rendering*, 2010, pp. 73–80.
[28] B. Gooch, G. Coombe, and P. Shirley, "Artistic vision: painterly rendering using computer vision techniques," in *Proc. Int'l symposium on Non-photorealistic animation and rendering*, 2002, pp. 83–ff.
[29] T. Wang, J. Collomosse, D. Slatter, P. Cheatle, and D. Greig, "Video stylization for digital ambient displays of home movies," in *Proc. Int'l symposium on Non-photorealistic animation and rendering*, 2010, pp. 137–146.
[30] A. Orzan, A. Bousseau, P. Barla, and J. Thollot, "Structure-preserving manipulation of photographs," in *Proc. Int'l symposium on Non-photorealistic animation and rendering*, 2007, pp. 103–110.
[31] J. E. Kyprianidis, H. Kang, and J. Döllner, "Image and video abstraction by anisotropic kuwahara filtering," in *Computer Graphics Forum*, vol. 28, no. 7, 2009, pp. 1955–1963.
[32] Y.-W. Tai, J. Jia, and C.-K. Tang, "Soft color segmentation and its applications," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 29, no. 9, pp. 1520–1537, 2007.
[33] F. Pitie, A. C. Kokaram, and R. Dahyot, "N-dimensional probability density function transfer and its application to color transfer," in *Proc. IEEE Int'l Conf. Computer Vision*, vol. 2, 2005, pp. 1434–1439.
[34] T.-W. Huang and H.-T. Chen, "Landmark-based sparse color representations for color transfer," in *Proc. IEEE Int'l Conf. Computer Vision*, 2009, pp. 199–204.
[35] Y. Hwang, J.-Y. Lee, I. S. Kweon, and S. J. Kim, "Color transfer using probabilistic moving least squares," in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2014, pp. 3342–3349.
[36] R. Irony, D. Cohen-Or, and D. Lischinski, "Colorization by example," in *Eurographics Symp. on Rendering*, vol. 2, 2005.
[37] A. Levin, D. Lischinski, and Y. Weiss, "Colorization using optimization," in *ACM Trans. Graphics*, vol. 23, no. 3, 2004, pp. 689–694.
[38] Y.-C. Huang, Y.-S. Tung, J.-C. Chen, S.-W. Wang, and J.-L. Wu, "An adaptive edge detection based colorization algorithm and its applications," 2005, pp. 351–354.
[39] X. Liu, L. Wan, Y. Qu, T.-T. Wong, S. Lin, C.-S. Leung, and P.-A. Heng, "Intrinsic colorization," *ACM Trans. on Graphics (SIGGRAPH Asia 2008 issue)*, vol. 27, no. 5, pp. 152:1–152:9, December 2008.
[40] X. Lu, Z. Lin, X. Shen, R. Mech, and J. Z. Wang, "Deep multi-patch aggregation network for image style, aesthetics, and quality estimation," in *Proc. IEEE Int'l Conf. Computer Vision*, 2015, pp. 990–998.
[41] Y. C. Shih, S. Paris, C. Barnes, W. T. Freeman, and F. Durand, "Style transfer for headshot portraits," *Acm Transactions on Graphics*, vol. 33, no. 4, pp. 1–14, 2014.
[42] Y. Shih, S. Paris, F. Durand, and W. T. Freeman, "Data-driven hallucination of different times of day from a single outdoor photo," *Acm Transactions on Graphics*, vol. 32, no. 6, pp. 2504–2507, 2013.
[43] A. Selim, M. Elgharib, and L. Doyle, "Painting style transfer for head portraits using convolutional neural networks," *ACM Trans. Graphics*, vol. 35, no. 4, pp. 129:1–129:18, July 2016.
[44] A. Y. Yang, J. Wright, Y. Ma, and S. S. Sastry, "Unsupervised segmentation of natural images via lossy data compression," *Computer Vision and Image Understanding*, vol. 110, no. 2, pp. 212–225, 2008.
[45] J. Wang, Y. Jia, X.-S. Hua, C. Zhang, and L. Quan, "Normalized tree partitioning for image segmentation," in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2008, pp. 1–8.
[46] H. Mobahi, S. R. Rao, A. Y. Yang, S. S. Sastry, and Y. Ma, "Segmentation of natural images by texture and boundary compression," *Int'l Journal of Computer Vision*, vol. 95, no. 1, pp. 86–98, 2011.
[47] X. Z. Fern and C. E. Brodley, "Solving cluster ensemble problems by bipartite graph partitioning," in *Proc. Int'l Conf. Machine Learning*, 2004, p. 36.
[48] Z. Li, X.-M. Wu, and S.-F. Chang, "Segmentation using superpixels: A bipartite graph partitioning approach," in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2012, pp. 789–796.
[49] J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid, "Deep convolutional matching," *Arxiv*, vol. 1506.07656, 2015. [Online]. Available: http://arxiv.org/abs/1506.07656
[50] J. Yuan, D. Wang, and A. M. Cheriyadat, "Factorization-based texture segmentation," *IEEE Trans. on Image Processing*, vol. 24, no. 11, pp. 3488–3497, Nov 2015.
[51] G. H. Golub and C. F. Van Loan, *Matrix computations*. JHU Press, 2012, vol. 3.
[52] H. Zha, X. He, C. Ding, H. Simon, and M. Gu, "Bipartite graph partitioning and data clustering," in *Proc. IEEE Int'l Conf. Information and Knowledge Management*, 2001, pp. 25–32.
[53] D. Bruff, "The assignment problem and the hungarian method," *Notes for Math*, vol. 20, 2005.
[54] K. He, J. Sun, and X. Tang, "Guided image filtering," in *Proc. IEEE European Conf. Computer Vision*. Springer, 2010, pp. 1–14.
[55] T. Welsh, M. Ashikhmin, and K. Mueller, "Transferring color to greyscale images," *ACM Trans. Graphics*, vol. 21, no. 3, pp. 277–280, 2002.
[56] M. Rubinstein, D. Gutierrez, O. Sorkine, and A. Shamir, "A comparative study of image retargeting," in *ACM Trans. Graphics*, vol. 29, no. 6, 2010, p. 160.
[57] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, November 2012.
[58] C. Y. Ren, V. A. Prisacariu, and I. D. Reid, "gSLICr: SLIC superpixels at over 250Hz," *ArXiv e-prints*, September 2015.
[59] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. IEEE European Conf. Computer Vision*, 2016.
[60] S. Yang, G. Cheung, P. L. Callet, J. Liu, and Z. Guo, "Computational modeling of artistic intention: Quantify lighting surprise for painting analysis," in *International Conference on Quality of Multimedia Experience (QoMEX)*, June 2016, pp. 1–6.
[61] A. J. Champandard, "Semantic Style Transfer and Turning Two-Bit Doodles into Fine Artworks," *ArXiv e-prints*, March 2016.
[62] F. Pitie and A. Kokaram, "The linear monge-kantorovitch linear colour mapping for example-based colour transfer," in *Visual Media Production, 2007. CVMP 2007. 4th European Conference on*, Nov 2007, pp. 1–9.

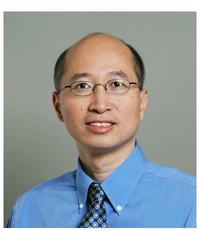
Jiaying Liu (08'S-10'M-17'SM) received the B.E. degree in computer science from Northwestern Polytechnic University, Xi'an, China, and the Ph.D. degree with the Best Graduate Honor in computer science from Peking University, Beijing, China, in 2005 and 2010, respectively.

She is currently an Associate Professor with the Institute of Computer Science and Technology, Peking University. She has authored over 100 technical articles in refereed journals and proceedings, and holds 20 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.

Wenhan Yang (17'S) received the B.S degree in Computer Science from Peking University, Beijing, China, in 2012. He is currently a Ph.D. student with the Institute of Computer Science and Technology, Peking University. Mr. Yang was a Visiting Scholar with the National University of Singapore, from 2015 to 2016. His current research interests include image processing, sparse representation, image restoration and deep learning-based image processing.

Xiaoyan Sun is a Lead Researcher with Microsoft Research Asia. Her current research interests include computer vision, image/video processing, and machine learning. She is now working on video analysis, image restoration, and image/video coding. Dr. Sun has authored or co-authored over 100 papers in journals and conferences, 10 proposals to standards with one accepted, and over 10 granted US patients. She was a recipient of the Best Paper Award of the IEEE Transactions on Circuits and Systems for Video Technology in 2009 and the Best Student Paper Award of VCIP 2016. Dr. Sun is a TC member of IEEE MSA (Multimedia Systems & Applications, AE of Signal Processing: Image Communication, and AE for the IEEE Journal on Emerging and Selected Topics in Circuits and Systems. She also served as session chair, area chair, and TC co-chair of several international conferences. Xiaoyan Sun received the B.S., M.S., and Ph.D. degrees in computer science from Harbin Institute of Technology, Harbin, China, in 1997, 1999, and 2003, respectively. She has been an intern in Microsoft Research Asia since 2000 and joined Microsoft Research Asia later in 2003. She is an Adjunct Professor (PhD supervisor) of the University of Science and Technology of China.



Wenjun (Kevin) Zeng (M'97-SM'03-F'12) is a Principal Research Manager and a member of the senior leadership team at Microsoft Research Asia. He has been leading the video analytics research empowering the Microsoft Cognitive Services and Azure Media Analytics Services since 2014. He was with Univ. of Missouri (MU) from 2003 to 2016, most recently as a Full Professor. Prior to that, he had worked for PacketVideo Corp., Sharp Labs of America, Bell Labs, and Panasonic Technology. Wenjun has contributed significantly to the development of international standards (ISO MPEG, JPEG2000, and OMA). He received his B.E., M.S., and Ph.D. degrees from Tsinghua Univ., the Univ. of Notre Dame, and Princeton Univ., respectively. His current research interest includes mobile-cloud media computing, computer vision, social network/media analysis, and multimedia communications and security.

He is an Associate Editor-in-Chief of IEEE Multimedia Magazine, and was an AE of IEEE Trans. on Circuits & Systems for Video Technology (TCSVT), IEEE Trans. on Info. Forensics & Security, and IEEE Trans. on Multimedia (TMM). He was a Special Issue Guest Editor for the Proceedings of the IEEE, TMM, ACM TOMCCAP, TCSVT, and IEEE Communications Magazine. He was on the Steering Committee of IEEE Trans. on Mobile Computing and IEEE TMM. He served as the Steering Committee Chair of IEEE ICME in 2010 and 2011, and is serving or has served as the General Chair or TPC Chair for several IEEE conferences (e.g., ICME'2018, ICIP'2017). He was the recipient of several best paper awards (e.g., IEEE VCIP'2016, IEEE ComSoC MMTC 2015 Best Journal Paper, ACM ICMCS'2012). He is a Fellow of the IEEE.