SOFT SEGMENTATION-GUIDED BIPARTITE GRAPH IMAGE STYLIZATION

Saboya Yang[1], *Jiaying Liu*[1*], *Wenhan Yang*[1], *Shuai Yang*[1], *Chunpeng Li*[2]

[1]Institute of Computer Science and Technology, Peking University
[2]Institute of Computing Technology Chinese Academy of Sciences

ABSTRACT

In this paper, we propose a photo stylistic brush, an automatic robust style transfer approach based on soft segmentation-guided bipartite graph. A two-step bipartite graph algorithm with different granularity levels is employed to aggregate pixels into superpixel 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 soft segmentation-guided bipartite graph, and superpixel-level correspondences are generated by a bipartite matching. Finally, the refined correspondence guides our approach to perform the transfer 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, soft-segmentation, bipartite graph, stylistic brush.

1. INTRODUCTION

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.

*Corresponding author. This work was supported by National Natural Science Foundation of China under contract No. 61471009, National Defense Innovation Foundation of the Chinese Academy of Sciences, Grant No. CXJJ-15M022 and the Key Research Program of the Chinese Academy of Sciences, Grant No. KFZD-SW-407.

Some works investigate image stylization by considering the style composition instead of a single style element. Most of these methods [11–13] devote to separating and dealing with the content and style individually. These methods suffer from two limitations: 1) The assumption that the content and the style could be separable may be questional. 2) From the application aspect, for photos, color, light, contrast *etc.* need to be paid more attention rather than painting styles, like textures and strokes. There are also recent works that regard the style transfer as adaptive local mappings [14–16]. They focus on addressing the local style transfer on a specific category - the facial image [15, 16] with assumed face related priors or by utilizing external coupled time-lapse videos [14] to create the local transfer mapping.

In this paper, we aim to solve the style transfer in the general case and 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. The proposed stylistic brush is realized by a robust style transfer method based on the soft segmentation-guided bipartite graph 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 soft-segmentation-guided bipartite graph to refine the final superpixel-level correspondent relationship. Finally, the proposed method transfers colors within each superpixel correspondence in a decorrelated color space to achieve the stylization. To the best of our knowledge, among natural photo style transfer approaches, it is the first attempt to handle the style transfer without the limitation on the specific category of input and reference images and not using crafted external priors. Extensive experiments demonstrate that our method significantly outperforms previous methods in the general style transfer.

2. SOFT SEGMENTATION-GUIDED BIPARTITE GRAPH FOR PHOTO STYLE TRANSFER

The proposed method transfers the style of the reference image to the input image by a two-step bipartite graph framework as shown in Fig. 1. It first detects the dense correspondence and calculates the designed hierarchical features. Based on the correspondence and features, our method then aggregates pixels into superpixels using a simple clustering algorithm for the pixels around the matched points and a bipartite graph framework for the pixels far from the matched points. Afterwards, the proposed approach transfers the colors be-

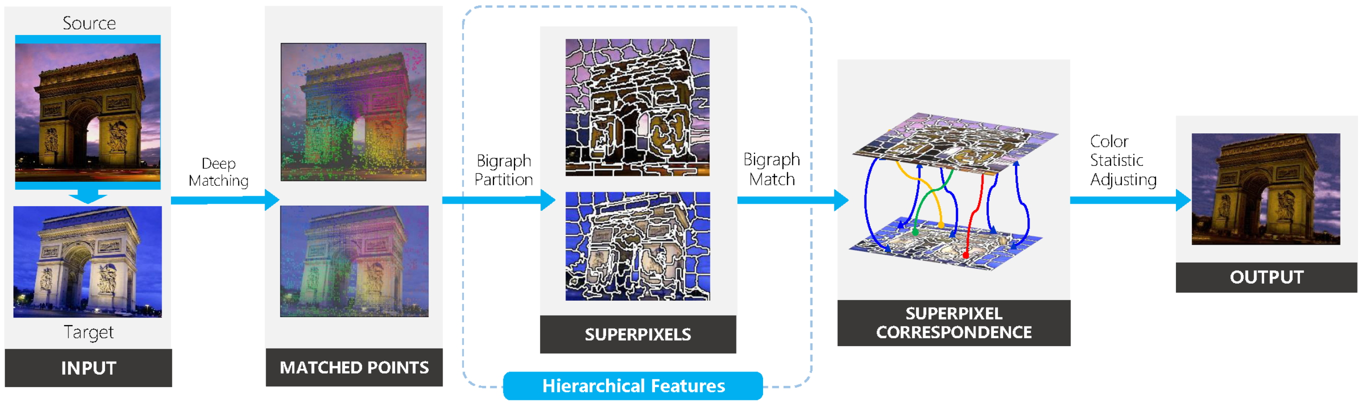


Fig. 1. The flowchart of the proposed soft segmentation-guided bipartite graph image stylization.

tween corresponding superpixels in a decorrelated color space.

2.1. Soft Segmentation Aggregation with Hierarchical Features

Superpixel is a pixel cluster consisting of several pixels with similar color and brightness, usually provided by an initialization for segmentation [17–19] or a soft constraint on segmentation [20, 21]. Compared with raw pixels, superpixel provides more reliable and fine-grained regions in comparison with segmented objects.

Our method 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 [22]. 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 $\mathbf{M}_{(i,j)}$;
- Color $\mathbf{C}_{(i,j)}$ at pixel (i, j);
- Gradient of a patch $\mathbf{DV}_{(i,j)}$;
- Absolute location $\mathbf{L}^{a}_{(i,j)}$;
- Texture feature $\mathbf{T}_{(i,j)}$ of a patch centered at pixel (i, j). The features of factorization-based texture segmentation [23] are extracted to segment different texture regions and locate their boundaries.
- Relative location, $\mathbf{L}^{r}_{(i,j)}$. 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).
- Locality-constrained linear coding (LLC) feature, $\mathbf{S}_{(i,j)}$. 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.

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. The proposed method further deals with other unsettled pixels in a bipartite graph framework hereafter.

2.2. Pixel Bipartite Graph Partition

After obtaining the superpixel around matched points, our approach 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, the proposed method 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,

$$\mathbf{J}\mathbf{g} = \lambda\mathbf{D}\mathbf{g}, \tag{4}$$

where λ is the eigenvalue to be optimized. And $\mathbf{J} = \mathbf{D} - \mathbf{\Omega}$ is the Laplacian matrix and $\mathbf{D} = diag(\mathbf{\Omega}\mathbf{1})$ 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 [24] on the normalized affinity matrix $\tilde{\mathbf{\Omega}} = \mathbf{D}^{-1/2}\mathbf{\Omega}\mathbf{D}^{1/2}$ or partial SVD [25] on normalized across-affinity matrix. We adopting the latter solution, and 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 $\mathbf{F}^r = [\mathbf{F}^{r,m}, \mathbf{F}^{r,u}]$.

2.3. Soft Segmentation-Guided Bipartite Graph Matching

In the above step, our method 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.

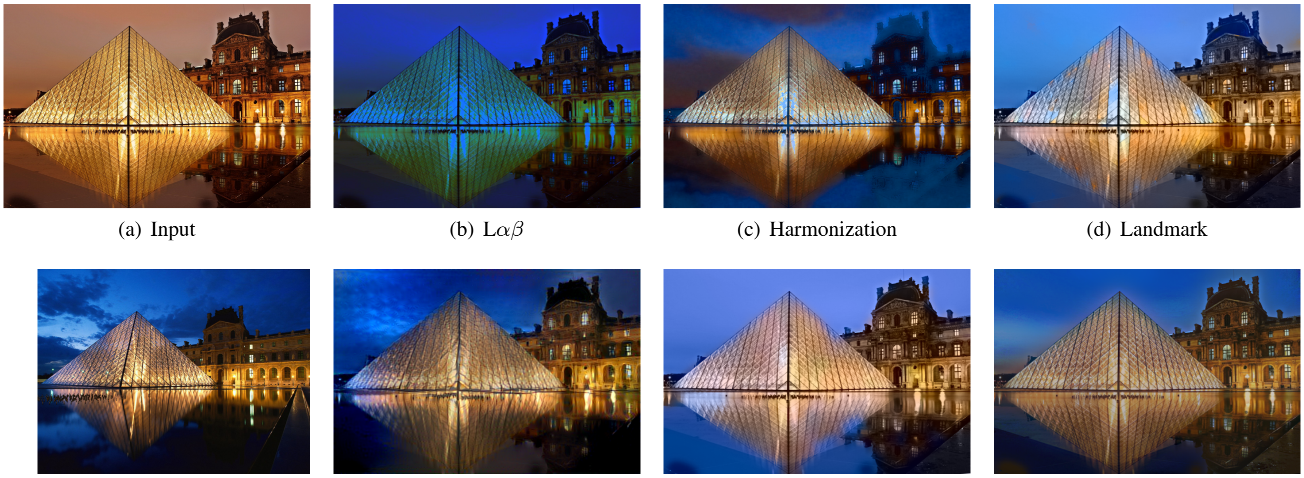
Thus, our approach constructs a superpixel bipartite graph and performs a graph matching on it. The nodes of the new graph represent superpixels of $\mathbf{F}^c$ and $\mathbf{F}^r$. There is an edge connection between corresponding nodes, only when their hierarchical features are close enough in the 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, the proposed method solves the bipartite graph matching by the Hungarian algorithm [26], obtaining final superpixel correspondences $\mathbf{F}^c_f$ and $\mathbf{F}^r_f$.

2.4. De-Correlated Style Transfer

After obtaining a reliable superpixel correspondence, we fit the color statistic of the input image into that of the reference one. Based on the proposed framework, the styles of an image could be transferred locally at the granularity of superpixel. Our method transfers colors by manipulating the statistic in the $l\alpha\beta$-CIE space, a de-correlated color space [1], as our local mapping method.

3. EXPERIMENTAL RESULTS

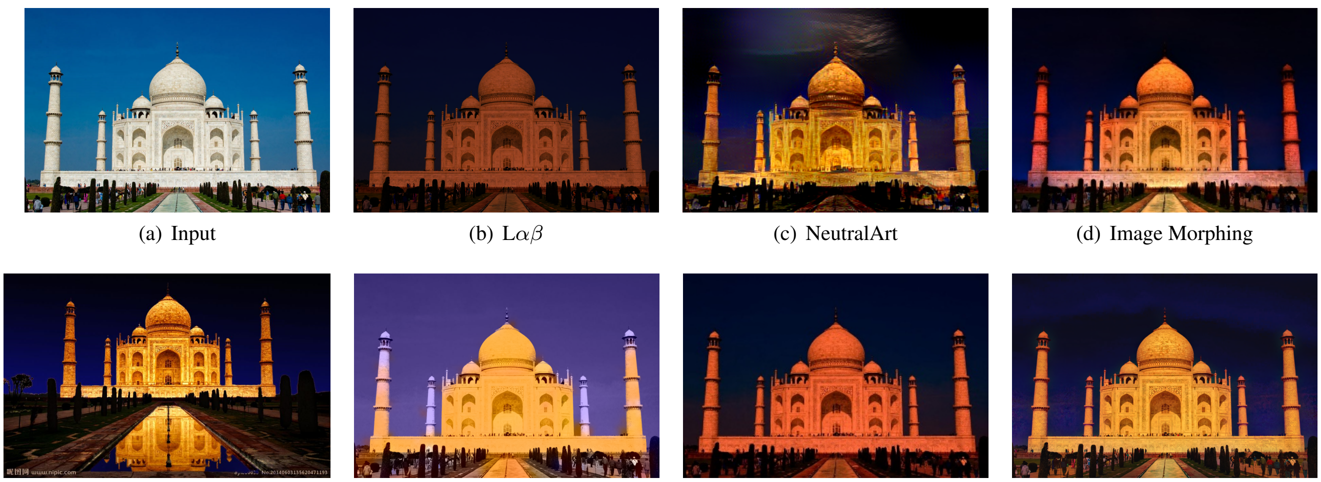
We compare the proposed method with the following eight state-of-the-art style/color transfer methods: L$\alpha\beta$ decorrelated color space (L$\alpha\beta$) [1], color "mood" transfer (MoodTrans) [27], multiscale harmonization (Harmonization) [7], landmark sparse color representation (Landmark) [28], neural algorithm of artistic style (NeutralArt) [13], superpixels matching (SuperMatch) [29], image morphing + SITF flow (Image Morphing) [15] and data-driven hallucination (Data-driven) [14]. To make a fair comparison in our case, for image morphing + SITF flow, 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, our method first turns the input image into greyscale one, then colorizes the generated greyscale image. We set the parameters as: $\lambda_{\mathbf{M}} = 0.1, \lambda_{\mathbf{T}} = 0.001, \lambda_{\mathbf{C}} = 0.0001, \lambda_{\mathbf{DV}} = 10^{-6}, \lambda_{\mathbf{S}} = 0.1, \lambda_{\mathbf{L^a}} = \lambda_{\mathbf{L^r}} = 0.01, n_{\boldsymbol{\alpha}} = 1000$ and $n_{\boldsymbol{\beta}} = 10^6$. These parameters are initialized as $\lambda_M = 0.01\lambda_T = 1, \lambda_C = 0.01, \lambda_D V = 0.01, \lambda_S = 1, \lambda_L a = \lambda_L r = 1$.



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

(e) Reference (f) NeutralArt (g) SuperMatch (h) Proposed

Fig. 2. Visual comparisons of style transfer among different algorithms on *Louvre* image pairs.



(a) Input (b) L$\alpha\beta$ (c) NeutralArt (d) Image Morphing

(e) Reference (f) SuperMatch (g) Data-driven (h) Proposed

Fig. 3. Visual comparisons of style transfer among different algorithms on *Taj Mahal* image pairs.

The comparison results of our method and other state-of-the-art methods for three input images are presented in Figs. 2 and 3. Please enlarge and view these figures on the screen for better comparison. The subjective quality of these results demonstrates the superiority of the proposed method. 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 Fig. 2(b), 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). For image morphing + SITF flow [15], as shown in Fig. 3(d), 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 [14], as shown in Fig. 3(g), 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 and effective superpixel bipartite framework for modeling in the global and local correspondences, our approach transfers the proper styles for the local regions in the generated results as shown in Figs. 2(h) and 3(h).

4. CONCLUSION

In this paper, we first introduce the concept of image stylistic brush and accordingly design an exemplar-based photo stylization method 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 superpixels first. Then, we generate a soft segmentation-guided bipartite graph, producing correspondences of the superpixels by bipartite matching. The correspondence is then used to guide the style transfer in a decorrelated color space. Extensive experimental results demonstrate that the proposed method achieves superior visual quality compared to state-of-the-art methods while providing style consistent with the reference image.

5. REFERENCES

[1] Erik Reinhard, Michael Ashikhmin, Bruce Gooch, and Peter Shirley, "Color transfer between images," *IEEE Computer graphics and applications*, , no. 5, pp. 34–41, 2001.

[2] Baoyuan Wang, Yizhou Yu, Tien-Tsin Wong, Chun Chen, and Ying-Qing Xu, "Data-driven image color theme enhancement," in *ACM Trans. Graphics*, 2010, vol. 29, p. 146.

[3] Tania Pouli and Erik 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] François Pitié, Anil C Kokaram, and Rozenn Dahyot, "Automated colour grading using colour distribution transfer," *Computer Vision and Image Understanding*, vol. 107, no. 1, pp. 123–137, 2007.

[5] Soonmin Bae, Sylvain Paris, and Frédo Durand, "Two-scale tone management for photographic look," in *ACM Trans. Graphics*, 2006, vol. 25, pp. 637–645.

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

[7] Kalyan Sunkavalli, Micah K Johnson, Wojciech Matusik, and Hanspeter Pfister, "Multi-scale image harmonization," in *ACM Trans. Graphics*, 2010, vol. 29, p. 125.

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

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

[10] Cassidy J Curtis, Sean E Anderson, Joshua E Seims, Kurt W Fleischer, and David H Salesin, "Computer-generated watercolor," in *Proc. Conf. on Computer graphics and interactive techniques*, 1997, pp. 421–430.

[11] Aaron Hertzmann, Charles E Jacobs, Nuria Oliver, Brian Curless, and David H Salesin, "Image analogies," in *Proc. Int'l Conf. on Computer graphics and interactive techniques*, 2001, pp. 327–340.

[12] Wei Zhang, Chen Cao, Shifeng Chen, Jianzhuang Liu, and Xiaoou Tang, "Style transfer via image component analysis," *IEEE Trans. on Multimedia*, vol. 15, no. 7, pp. 1594–1601, 2013.

[13] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge, "A neural algorithm of artistic style," *Arxiv:*, vol. abs/1508.06576, 2015.

[14] Yichang Shih, Sylvain Paris, Frédo Durand, and William 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.

[15] Yi Chang Shih, Sylvain Paris, Connelly Barnes, William T. Freeman, and Frédo Durand, "Style transfer for headshot portraits," *Acm Transactions on Graphics*, vol. 33, no. 4, pp. 1–14, 2014.

[16] M. Elgharib A. Selim and L. Doyle, "Painting style transfer for head portraits using convolutional neural networks," 2016.

[17] Allen Y Yang, John Wright, Yi Ma, and S Shankar Sastry, "Unsupervised segmentation of natural images via lossy data compression," *Computer Vision and Image Understanding*, vol. 110, no. 2, pp. 212–225, 2008.

[18] Jingdong Wang, Yangqing Jia, Xian-Sheng Hua, Changshui Zhang, and Long Quan, "Normalized tree partitioning for image segmentation," in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[19] Hossein Mobahi, Shankar R Rao, Allen Y Yang, Shankar S Sastry, and Yi Ma, "Segmentation of natural images by texture and boundary compression," *Int'l Journal of Computer Vision*, vol. 95, no. 1, pp. 86–98, 2011.

[20] Xiaoli Zhang Fern and Carla E Brodley, "Solving cluster ensemble problems by bipartite graph partitioning," in *Proc. Int'l Conf. Machine Learning*, 2004, p. 36.

[21] Zhenguo Li, Xiao-Ming Wu, and Shih-Fu Chang, "Segmentation using superpixels: A bipartite graph partitioning approach," in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2012, pp. 789–796.

[22] Jérome Revaud, Philippe Weinzaepfel, Zaïd Harchaoui, and Cordelia Schmid, "Deep convolutional matching," *Arxiv*, vol. 1506.07656, 2015.

[23] 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.

[24] Gene H Golub and Charles F Van Loan, *Matrix computations*, vol. 3, JHU Press, 2012.

[25] Hongyuan Zha, Xiaofeng He, Chris Ding, Horst Simon, and Ming Gu, "Bipartite graph partitioning and data clustering," in *Proc. IEEE Int'l Conf. Information and Knowledge Management*, 2001, pp. 25–32.

[26] Derek Bruff, "The assignment problem and the hungarian method," *Notes for Math*, vol. 20, 2005.

[27] Tomihisa Welsh, Michael Ashikhmin, and Klaus Mueller, "Transferring color to greyscale images," *ACM Trans. Graphics*, vol. 21, no. 3, pp. 277–280, 2002.

[28] Tzu-Wei Huang and Hwann-Tzong Chen, "Landmark-based sparse color representations for color transfer," in *Proc. IEEE Int'l Conf. Computer Vision*, 2009, pp. 199–204.

[29] Raj Kumar Gupta, Alex Yong-Sang Chia, Deepu Rajan, Ee Sin Ng, and Huang Zhiyong, "Image colorization using similar images," 2012, pp. 369–378.