

Pixel-Based Impressionistic Stylization

Hochang Lee¹, Dongwann Kang², Taemin Lee³ and Kyunghyun Yoon³

¹*Agency for Defense Development,
Rep. of Korea hcllee0126@add.re.kr*

²*Bournemouth University, United Kingdom dkang@bournemouth.ac.uk*

³*Chung-Ang University, Rep. of Korea (kevinlee,khyoon) @cglab.cau.ac.kr*

Abstract

In this paper, we propose a novel pixel-based stylization technique that analyzes and transfers impressionistic styles. A pixel-based stylization technique has tended to focus on only a single term (color or texture), so that it is not easy for them to express complex styles, such as impressionist paintings. In this paper, we propose a novel pixel-based stylization technique that analyze and transfer impressionistic styles. We analyze regions of a source image with respect to three characteristics – color pattern, texture, and flow pattern – and then transfer each region's impressionistic style to a target image based on saliency similarities.

Keywords: *Style Transfer, Non-photorealistic Rendering, Painterly Rendering*

1. Introduction

Painting expresses a scene on canvas using brush strokes and pigments, with various textures and colors depending on the brushes. A wide variety of styles of painting have emerged nowadays, many of which are found to depend on the type of brush, the method of paint application, and the mode of color mixture on the canvas. In the 19th century, Impressionist painters made innovative use of color contrast, brush orientation, and texturing to achieve a novel style from previous academism painters who followed standard rules. Figure 1 provides a famous example of such expression, a self-portrait by Vincent Van Gogh. In this image, we can see that the background is expressed using complementary contrastive effects, mixing reddish and bluish hues, while brush stroke orientation is arranged according to the outline of the depicted figure. The figure itself, on the other hand, is expressed using less contrastive yellowish hues, and brush stroke orientation follows the normal shape (spreadable effects).

In the late 1990's, researchers began to explore new rendering techniques for transferring painterly styles from a reference image to a target image. In these reference-based style transfer techniques, either the color or the texture of the reference image was analyzed numerically and applied in some way to the target image. However, there have been very little research considering both color and texture information. Also, most of it has been limited to simple patterns of color or texture. Thus, no such technique has yet been proposed to successfully transfer the rich characteristics (color contrast, stroke direction pattern, etc.) of impressionist painting.

In this paper, we propose a pixel-based style transfer technique that can successfully transfer the style of a reference impressionist painting to a target image. To accomplish this, we shall segment the reference image and analyze two different style components: color contrast and intensity of texture. We then apply these same style components to regions of the target image automatically. To match regions of the reference and target images, we make use of image saliency information, which helps discriminate what portions of the image are noticeable to the human eye. We segment the target image and extract the saliency distribution of each region. Based on the saliency distribution of the

region in which a pixel located, we determine a candidate set of pixels from the source image and then select the best candidate pixel from this set based on extracted style information. From our technique, we can transfer the color contrast of the reference image to the target image based on matched regions.

Our work presents the following contributions. First, we extended existing texture transfer algorithms by considering not only low-level features but also high-level features, allowing us to express impressionistic texturing effects more fully. Second, we achieved a more appropriate stylistic transfer by using saliency information to identify corresponding regions in both target and reference images. Finally, we tried to evaluate our transfer technique through a quantitative test.

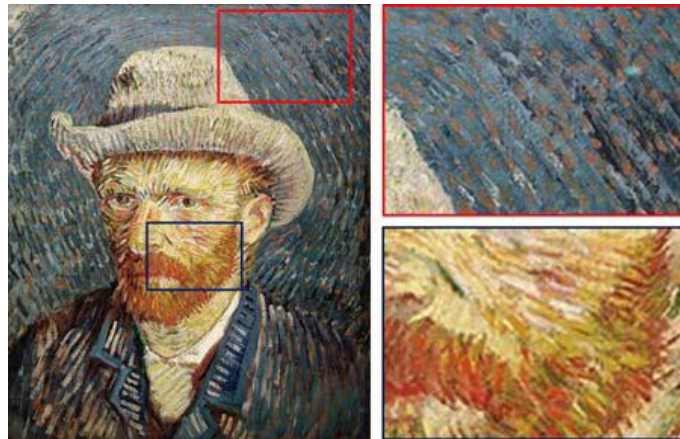


Figure 1. Real Painting (Vincent Van Gogh, Self-Portrait), in the Red Color Box, We Can See Complementary Contrastive Effects and Coherent Direction Flow According to Outline. In the Blue Color Box, We Can See Similar Colors are used and Stroke Direction Follows the Normal Shape of the Face

2. Related Work

The use of painterly rendering techniques for expressing realistic painting effects in computer graphics was first proposed by [7] and carried further in the work of [10] and [23]. Papers by [9] and [11] suggested a method for expressing stroke textures using texture mapping. [17, 26, 31, 35] further suggested simulating the styles of specific painters such as Van Gogh, Seurat, *etc.* [14] and [34] discussed controlling various stroke effects based on properties of the painted (*i.e.* rendered) object. All of these research activities belong to the field of stroke-based rendering [12], which expresses painting effects using the stroke as a basic unit. Stroke-based rendering's advantage is that the result can be modified by transforming the size, direction, or shape of the strokes. However, some trial and error is needed to obtain a result exhibiting the effect of the reference image.

To better express the style of a painted reference image, style transfer techniques have also emerged as a topic of research. These techniques can generally be divided into two groups: those that transfer texture characteristics, and those that transfer color characteristics. Texture transfer was derived from texture synthesis research described in [1, 3, 22]. [13] introduced the image analogies technique for expression of an artistic effect using texture transfer. This method produces impressive results using globally optimal and locally coherent texture synthesis, but it is also time-consuming and requires that the user enter the non-filtered image as an input. To overcome this limitation, [2] proposed the determination of a candidate set based on local search and the extraction of results by looking at the mean and standard

deviation of luminance data. This is less time-consuming and does not require user input of the non-filtered image. [21] uses an additional direction factor in the distance function to make use of image shape in pixel-based rendering, producing an artistic effect that can clearly express shape information. Though these methods do a good job reflecting the texture pattern of the source image, they cannot reflect the flow of stroke and color information of a reference image.

It should be noted that all of the above research focuses on either color or texture, but not both. In [6] and [13], both color and texture are considered, but the work is primarily aimed at texture synthesis and tries to synthesize color and texture naturally with neighboring pixels or patches. Thus, they do not consider color contrast in the context of artistic character. In [24, 28, 30], color information is transferred from a reference image based either on histogram information or average and standard deviation information. Unlike these, our algorithm transfers color and texture at the same time using actual pixel values of the reference image directly. Recently, [25] has suggested a style transfer method that considers color. In his paper, he expresses color and texture information using color patches derived from the source and target images. In contrast to their work, our technique introduces saliency information to the pixel-based approach.

Saliency information is now widely used in computer graphics. [18] and [20] have used saliency for the visualization of volume and mesh data. [4] has described image segmentation based on saliency values. For painterly effects, [5] has suggested controlling various attributes of brush strokes using saliency values. It should be noted that in all of these previous investigations, saliency techniques have been applied to stroke-based methods of rendering. No work has yet been done on the use of saliency in the context of pixel-based texture transfer.

Evaluating methods for NPR is currently challenging work. In stroke-based rendering, [15] and [27] have proposed an evaluation method for quantitatively estimating the temporal and time coherence value of their results. In general, quantitative evaluations of NPR reduce to algorithm performance in terms of time and memory cost, but for estimating the aesthetic similarity between a reference image and a resulting image, the best methods to date are user surveys and other forms of feedback. In this paper, we attempt to address this by suggesting and using a quantitative method of evaluation based on image retrieval techniques. Using this method, we hope to determine whether a style transfer has succeeded, and whether greater stylistic similarity was achieved compared to previous texture and color transfer techniques.

3. Style Analysis from Reference Image

There are many visual characteristics that define the style of an impressionist painting. In this paper, we select and analyze two basic characteristics: color and texture. For some impressionistic paintings, these characteristics may vary significantly even within the same image. To properly express this variation, we have performed a mean-shift segmentation of the reference image and collected separate style information for each region. We also extracted saliency information based on [8]'s saliency extraction technique. Paintings are a mixture of color and texture information, so it is necessary to separate color and texture information for effective analysis. To accomplish this separation, we use [33]'s image decomposition approach, and divide into high-frequency image information (texture) and low-frequency image information (color). Let S be the reference (source) image and each region of segment defined by S_n . Figure 2 shows the information extracted from a reference image and we define each value by S_{seg} , S_{sal} , S_{col} and S_{tex} . The reference image (S) is defined as follows:

$$S = \{s_1, s_2, \dots, s_n\}, \quad n = \text{number of segment} \quad (1)$$

$$S_n = \{Sal_n, Col_n, Flow_n, Tex_n\} \quad (2)$$

Where Sal_n means average saliency of n'th region and Col_n and Tex_n is style value of n'th region. Each style value is explained in the following section.



Figure 2. Extracted Information from a Reference (Source) Image

3.1. Color Style Analysis

To analyze color style, we considered the color distribution of each segment. For this, we extracted a hue distribution from S_{col} per region and selected the colors with highest frequency in the resulting histogram, defining c as the color with maximum frequency. Based on c , we then looked for the hue distance for other color values, and calculated the ratio of each color's portion of the high-frequency color space. Figure 3 shows the hue histogram for a region selected from background of Figure 1, and the constituent color's portions of the high-frequency color space. Finally, Col_n is calculated using equation (3)

$$Col_n = \{(c, p_1), (c + \theta_1, p_2), (c + \theta_2, p_3), \dots\} \quad (3)$$

Where c is the maximum frequency hue value, θ is the hue distance from c , and p is the proportion of the color to the high-frequency color space. The range for hue distribution is 1~360, and for p_n is 0~1.

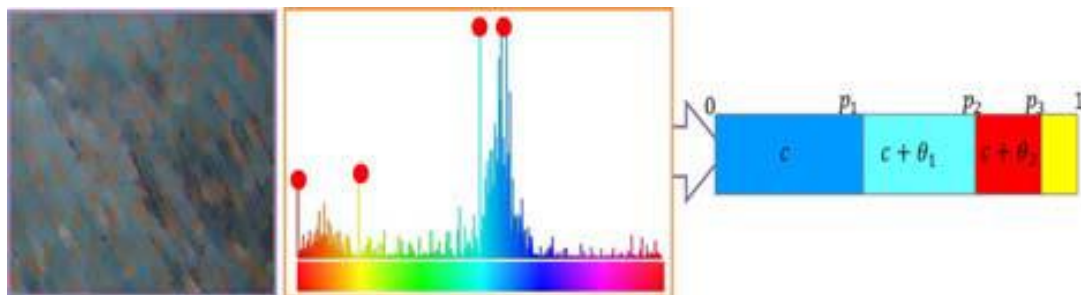


Figure 3. Color Analysis of a Reference Image Region

3.2. Texture Style Analysis

In real paintings, the texture effect is dependent on the type of brush and the amount of paint used in a region of brush strokes. To analyze texture style, we used S_{tex} . In S_{tex} , we can see the regions located brush have brighter intensity value than other regions. Thus, we can calculate the average luminance of a given region and, based on this value, analyze the intensity of the texture within each region.

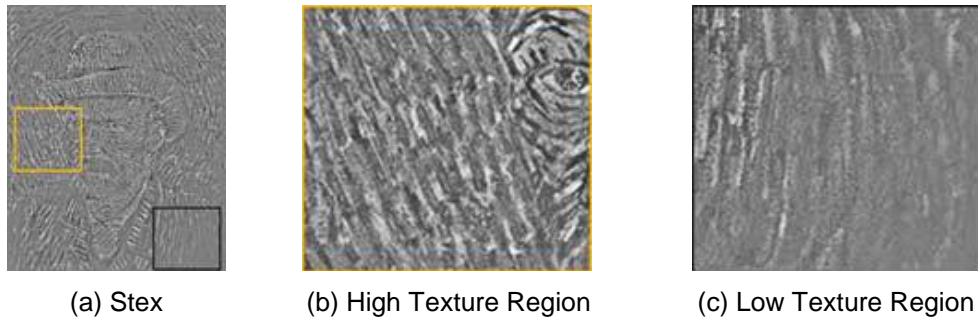


Figure 4. Texture Strength at Each Region

4. Style Transfer Based on Image Saliency

To transfer the style of a reference image, we modify [21]'s pixel based texture transfer algorithm and use it. This technique is very simple and time efficient. As in the reference analysis step, we again extract segment and saliency information from the target image (Figure 5 Below, we first provide a general outline of our algorithm, and then provide detail on specific techniques.)



Figure 5. Extracted Information from the Target Image

4.1. Outline of Our Style Transfer

Given target image T , reference image S , and result image R :

Initialize R by filling it with pixels taken randomly from S . Let function g maintain the relation between the mapping of pixels: pixel r in R obtains its value from pixel $g(r)$ in S .

Visit each pixel r in R , scanning in order from top-left to bottom-right. (a) Build candidate set Q based on image saliency (See Figure 6) (b) Select minimum distance argument from Q , and update r in R . (See Figure 7 and Figure 8)

Go back to step 2 if necessary (as in the case where the iteration parameter is greater than 1).

4.2. Style Matching between Reference and Target Image

For style matching between each segment of T_{seg} and S_{seg} , we use image saliency information. We find average saliency value per each segment region of T , and find matching regions of S with similar Sal_n . From this step, the background region of T with low saliency value is automatically matched with the background region of S . However, some regions have unnatural matched backgrounds. For example, the highest saliency region in T is face region. On the other hand, the highest saliency region in S is neck region. Accordingly, we just use saliency based matching by guiding information, and the final matching decision is determined by the user.

4.3. Candidate Set Construction Based on Image Saliency

It is important to construct the candidate pixel set in a way that accurately reflects the texture and color of the reference image without costing too much processing time. To help achieve this accuracy, our technique makes use of saliency information to construct a candidate pixel set for a given pixel r . There are two reasons for using saliency in this way. First, within a group of similar paintings, there tends to have similar saliency distribution. Second, when we perform style transfer, we are most concerned with moving a style from a highly focused region on the reference image to a highly focused region on the target image. We classify all pixels in the source image on the basis of their saliency value. We then extract the saliency range of the region in the target image where the current pixel r lies. The candidate set is composed of k source image pixels that lie within the saliency distribution range of the target image, with the value of k chosen by the user. In general, we use $k=15$. Figure 6 shows the process of selecting the candidate set based on saliency value.

4.4. Distance Function

We use equation (4) and (5) to calculate the similarity between each candidate pixel q in a candidate set Q and the current pixel r . We then update the value of r to the value of the candidate pixel that lies at minimum distance from r . The distance function is composed of from 3 terms: the color term, the direction term, and the texture term. This three-term composition is similar to [21]'s distance term; however we modify each term by considering the style of the reference image,

$$\text{Update } r = \arg \min_{q \in Q} D(r, q) \quad (4)$$

$$D(r, q) = W_c \cdot D_{col}(r, q) + D_{tex}(r, q) + W_d \cdot D_{dir}(r, q) \quad (5)$$

Where the W_c and W_d is color and direction weight and we generally use $W_c=1$ and $W_d=0.5$.

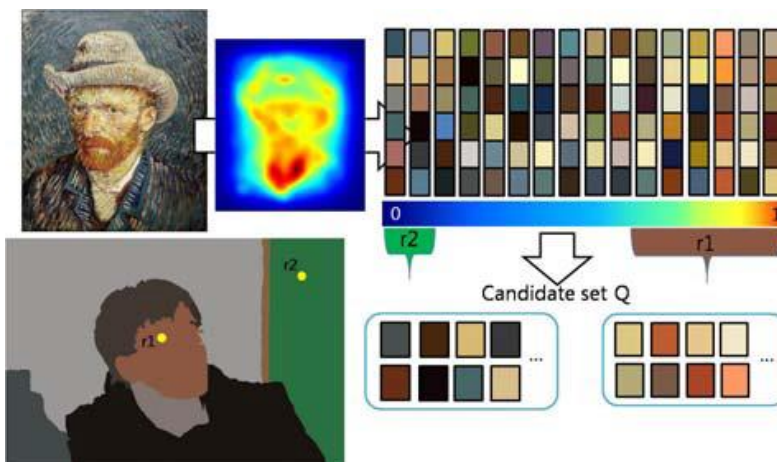


Figure 6. Selecting the Candidate Set Based on Saliency Value and Segmented Region Information

4.4.1. Color Term

The color distance term is calculated based on luminance and hue (equation 6~8):

$$D_{col}(r, q) = \|T_r - q_v\|_2 + \|\varepsilon - q_h\|_2 \quad (6)$$

$$\overline{T_r} = T(x), x \in \Phi(r) \quad (7)$$

$$\varepsilon = \begin{pmatrix} c & \text{if } \omega < p_1 \\ c + \theta_1 & \text{elseif } p_1 < \omega < p_2 \\ c + \theta_2 & \text{elseif } p_2 < \omega < p_3 \\ \dots & \dots \end{pmatrix} \quad (8)$$

Where Tr is the luminance value of pixel r in the target image, and qv is the luminance value of the current candidate pixel. ε is changed hue value and qh is the hue value of the current candidate pixel, while ω is a random value between 0 and 1. Previous proposed techniques [2], [21] used the mean value of a square filter centered on the current pixel r ; however, the results of these techniques tended to look blurred because the square area averages out the luminance value. In this paper, we instead generated an anisotropic kernel and utilized the mean value of the corresponding filter. To achieve this, we extracted tensor information from the image using the edge tangent flow described in [16] and calculated the mean luminance of the area using Kuwahara filtering as suggested by [19]. We then compared the calculated anisotropic mean luminance value Tr and the candidate pixel luminance qv in equation (8). Figure 7 shows the tensor directions and Kuwahara filtering for an image. The second term of this equation denotes color similarity. We calculated ε based on hue distribution for the selected segment. We changed the current target pixel value based on the color distribution ratio p_n . This allows us to express various color styles on the target image even when a given region of the target image starts out with a solid color. Figure 8 shows the color transferring result according to color table.

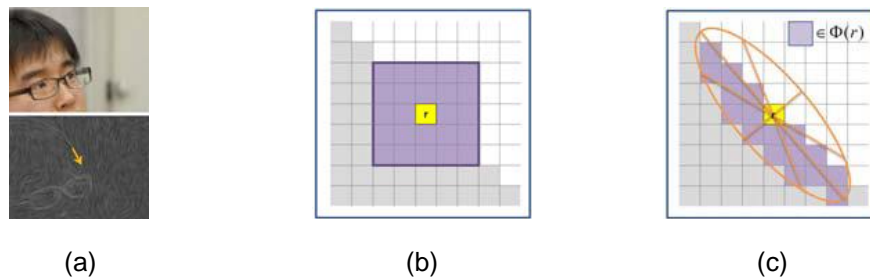


Figure 7. (a) Flow Field for a Selected Region (b), (c) Square and Kuwahara Filtering Based on Current Pixel Information

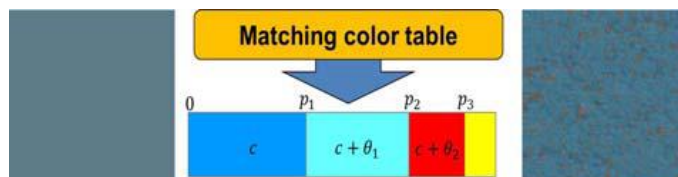


Figure 8. Color Transfer Step Based on Matching Color Table

4.4.2. Directional Term

We used the direction of the I-shape (proposed in [21]) according to the edge flow. Because of this term, the direction of each texture follows the gradient of source image. So that the results are seen naturally

$$D_I(r, q) = \left\| \overline{R(x_i)} - qv \right\|_2, x_i \in I(r) \quad (9)$$

Where $R(x_i)$ is the average luminance value of pixel x_i in the I-shape neighborhood of result image.

4.4.3. Texture Term

Our technique determines texture effect dynamically according to the neighbor size parameter. We calculate neighbor size based on the average intensity value of matched region S . The user only needs to set the minimum and maximum neighbor size. Here, we use a minimum size of 5 and a maximum size of 13.

4.5. Updating the Pixel

After identifying the candidate pixel at minimum distance from the target pixel, we update pixel r with the candidate pixel's hue and luminance value. We retain the saturation of the target image.

$$R(r) = \{q_k^{\wedge}, T(r), q_v^{\wedge}\} \quad (10)$$

5. Experimental Results

We tested our algorithm on various source images and compared it with other texture transfer methods. The algorithm takes about 20s to execute on a 2.8 GHz Pentium with 4 GB of memory for a HD size. Figure 9 shows the results when using Figure 2 as a reference image. In the face portion of the target image, there is a distinct yellowish texture similar to that of the reference image, and in the background there are color contrast patterns even in target regions that were solidly colored before the transfer. Judging from this result, our technique for style transfer is a success: multiple color and texture styles were transferred effectively to appropriate regions the target image. Furthermore, our use of an anisotropic kernel eliminated blurring artifacts.

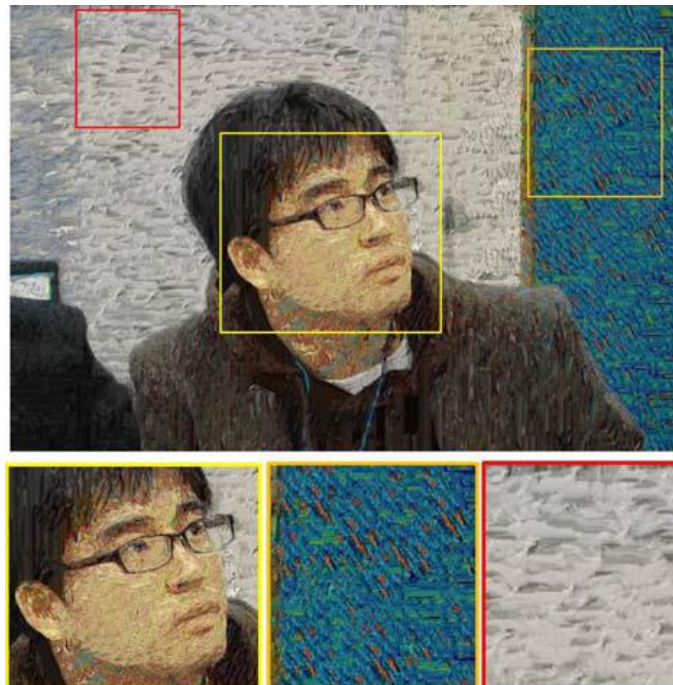


Figure 9. Resulting Image

Figure 10 compares our algorithm with other algorithms. Figure 10(a) shows the result of Ashkhmin's algorithm, and Figure 10(b) shows the result of applying Reinhard's color transfer algorithm and texture transfer algorithm simultaneously. Because previous texture transfer algorithms use only the color of the target image, the source image color is not reflected in the result. Figure 10(c) shows Lee's directional texture transfer technique. In this result, we can see the directional texture effect according to target flow. However, they also consider only luminance channel and use the original color of target image. Figure 10(d) shows our results without considering saliency. We eliminate blurred artifacts by using an anisotropic kernel. Also, note that in Figure 9 the texture effect appears stronger than in Figure 10(d). By choosing a candidate based on the pixels' saliency value, we create candidate sets of similarly valued pixels and generate better results. Figure 11 shows the results for various source images. Note that our algorithm reflects the color of the source image and not that of the target image. Figure 11 shows the result when we apply our technique using a reference image that is similar to our target image. We use the direction of the I-shape (proposed in [21]) according to the edge flow. Because of this term, the direction of each texture follows the gradient of source image. So that the results are seem naturally

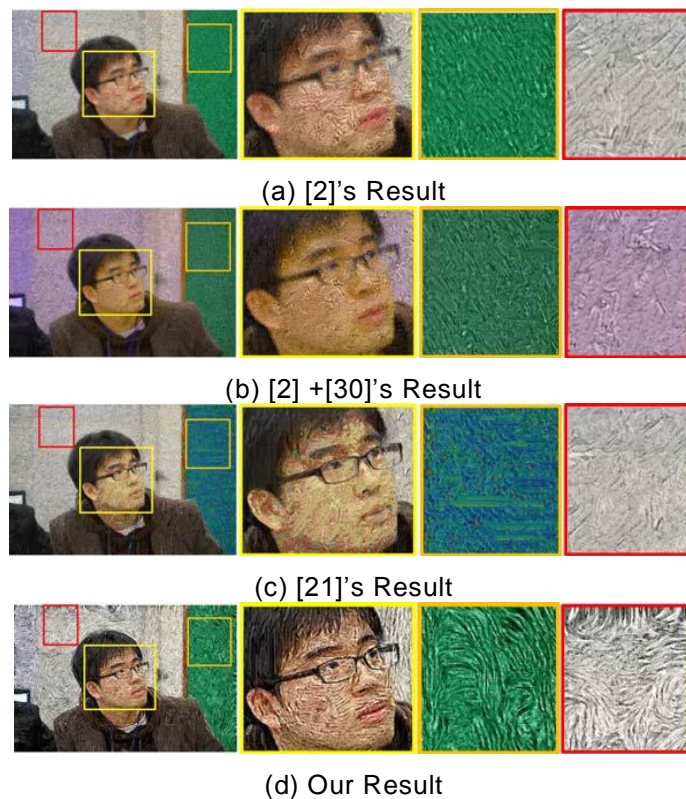


Figure 10. Result from Previous Works' Algorithm

6. Evaluation of Results

To evaluate our algorithm, we carried out both a quantitative and a qualitative test. In the quantitative test, we compared similarities between rendering result and reference image based on contents' similarity evaluation. In the qualitative test, we scored the feeling from the viewer after showing them all results of previous research algorithms.

6.1. Quantitative Test

The goal of style transfer is to transfer the characteristic style of a reference image to a target image. To estimate quantitatively how successful this transfer is (*i.e.* how stylistically similar the resulting image is to the reference image), we utilize methods developed in work on content-based image retrieval (CBIR). The overall aim of CBIR is to find the best-matched image from a DB given a user designated input image. CBIR queries are based on image content information like texture and color, and as such provide a convenient method for estimating the similarity of our reference and result images. We apply this method to evaluate our own transfer technique, as well as four others. To find specific information on color and texture, we divide the image between color and texture data using image decomposition technique. We then evaluate color and texture separately. To measure color similarity, we use the color NxM grams algorithm [32], outlined as follows: simplify by k according to hue value (generally, k=6). Analyze the pattern in the NxM grid and look for frequencies. Based on the frequency histograms of the result and reference images, find the region of intersection. The size of the region of intersection is determined by equation (11), yielding a value between 0 and 1. The closer the value is to 0, the greater the region of intersection and the more similar the two images are with respect to color.

$$Col_{sim}(Q, I) = 1 - \frac{2 * \sum_{j=1}^T \min(t(j, Q), t(j, I))}{\sum_{j=1}^T t(j, q) + \sum_{j=1}^T t(j, I)} \quad (11)$$

Where Q indicates the reference image, I indicates the result image, T represents the total pattern that results from the NxM grid (calculated as $k^{(N*M)}$) and t indicates the frequency of the current pattern j. To measure texture similarity, we use the GLCM algorithm. This algorithm extracts four features – contrast, energy, entropy and homogeneity – from a given texture. Each detail algorithm and equation is explained in [29]. We calculated feature value of reference image and each result. We then attempted to find the distance value of each feature. Based on these two methods, we tested 20 sample images. We generated results by ours and previous pixel based algorithm and calculated average distance value. We were then able to get the following results (Table 1).

From these results, we can see that our algorithm achieves higher general similarity than other algorithms. Because our algorithm uses hue values of the reference image directly, it produces a much smaller color distance than other algorithms that use average and stand deviation of whole color. This is therefore not good in painting, which has distinct color at each region. As for texture similarity, our algorithm again shows a smaller overall distance value than other algorithms, though we do get a larger distance for the contrast feature compared to the [21] algorithm. We estimate our algorithm can transfer style of reference image well.

Table 1. Comparing Results

Factor	Our	Lee10's	Ashikhmin03's	Reinhard01's	Reinhard01's+Ashikhmin03's
Color	0.266	0.479	0.486	0.616	0.3616
Texture (Contrast)	0.050	0.041	0.064	0.149	0.063
Texture (Energy)	0.004	0.005	0.013	0.879	0.013
Texture (Homogeneity)	0.020	0.002	0.031	0.589	0.269

6.2. Qualitative Test

We wanted to know how much our results are similar to paintings. To achieve this, we used user study. The user sample consisted of 30 participants between 20 and 30 years of age, with various majors in undergraduate and graduate studies; 15 engineering majors, 5 art majors, and 10 other majors. Our study used two tests to compare the results of a conventional style transfer algorithm: a stroke based rendering algorithm, and our algorithm. We first asked the participants to assign a score ranging from 1 to 7 to indicate how natural a resulting image seemed. Next, we showed the reference image and resulting image together, and asked users to assign a score ranging from 1 to 7 to indicate the degree of similarity between the two images.

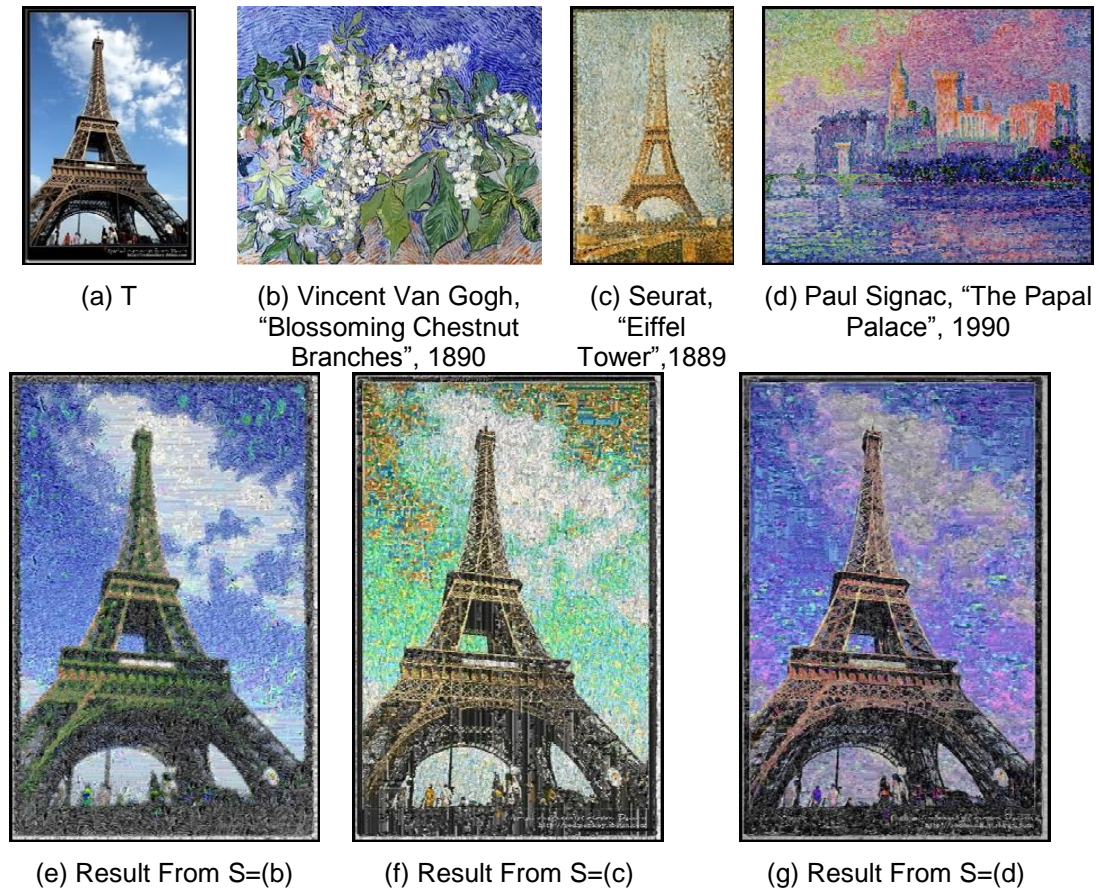


Figure 11. Various Results According To S

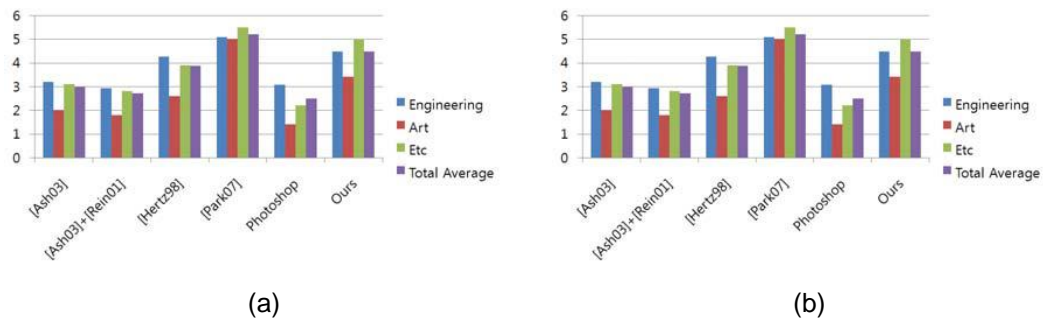


Figure 12. User Study Results

The results from the two tests are shown in Figure12. In first asking, as you can see in (a), stroke-based rendering techniques received a high score. Clear textures and various sizes of stroke caused this result. Our method got a lower score than stroke-based renderings. However, our results relatively got a higher score than other texture transfer methods because of various colors of brush stroke, texture, and considering object shape. In (b), texture transfer methods got a high score because stroke-based rendering did not reflect reference images. Our results got the highest score because of reflecting the color and texture of reference images.

7. Conclusions and Future Works

In this paper, we proposed a novel impressionist style transfer technique that improves upon previous style transfer techniques by taking into account both low-level and high-level features, such as color contrast and directional patterning, of the source image. To achieve this, we analyzed the color contrast, orientation and texture of regions per each segment and matched target segment based on image saliency. We also constructed candidate sets of pixels on the basis of target image saliency and tried to find the best candidate pixel by considering style information. Our algorithm sometimes generates unnatural results because our saliency information is not perfect and makes unnatural matching or candidate sets. However, we can almost successfully transfer the characteristic style of a source image to a target image in a single pass.

There are several opportunities for further research in this area. Currently, our algorithm does not consider the saturation of a candidate pixel, sometimes leading to unnatural results. In the sky part of Figure 13, for example, the saturation of T is high but the saturation of S is low, resulting in some loss of color style. To prevent this loss, we intend to integrate the saturation value into our distance function. We would also like to explore the addition of outline style to our current set of transferred styles. Finally, we are planning to reduce time cost by GPU based algorithm.



Figure 13. Limited Work(Sky Part)

Acknowledgments

This work was supported by the Chung-Ang University research grant in 2014 and the National Research Foundation of Korea(NRF) grant funded by the Korea government (MEST) (No. 2013R1A2A2A01069032).

References

- [1] M. Ashikhmin, "Synthesizing natural textures", In: In ACM Symposium on Interactive 3D Graphics, (2001), pp. 217–226.
- [2] M. Ashikhmin, "Fast texture transfer", IEEE Computer Graphics and Applications, vol. 23, no. 4, (2003), pp. 38–43.
- [3] J. S. D. Bonet, "Multi-resolution sampling procedure for analysis and synthesis of texture images", In: Proceedings of SIGGRAPH 1997, (1997), pp. 361–368.
- [4] M. M. Cheng, G. X. Zhang, N. J. Mitra, X. Huang and S. M. Hu, "Global contrast based salient region detection", In: IEEE CVPR, (2001), pp. 409–416.
- [5] J. P. Collomosse and P. M. Hall, "Painterly rendering using image salience", In: Proceedings of the 20th UK conference on Eurographics. EGUK '02, (2002), pp. 122–128.
- [6] A. A. Efros and W. T. Freeman, "Image quilting for texture synthesis and transfer", In: Proceedings of SIGGRAPH 2001, (2001), pp. 341–346.
- [7] P. Haeblerli, "Paint by numbers: Abstract image representations", In: Proceedings of SIGGRAPH 1990, (1990), pp. 207–214.
- [8] J. Harel, C. Koch and P. Perona, "Graph-based visual saliency", In: Advances in Neural Information Processing Systems 19, MIT Press, (2007), pp. 545–552.
- [9] J. Hays and I. Essa, "Image and video based painterly animation", In: Symp. Non-Photorealistic Animation and Rendering (NPAR2004), (2004), pp. 120–133.
- [10] A. Hertzmann, "Painterly rendering with curved brush strokes of multiple sizes", In: Proceedings of SIGGRAPH 1998, (1998), pp. 453–460.
- [11] A. Hertzmann, "Fast paint texture", In: In Second International Symposium on Non-Photorealistic Animation and Rendering (NPAR 2002), (2002), pp. 91–96.
- [12] A. Hertzmann, "A survey of stroke-based rendering", In: IEEE Computer Graphics and Applications, (2003), pp. 70–81.
- [13] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless and D. H. Salesin, "Image analogies", In: Proceedings of the 28th annual conference on Computer graphics and interactive techniques. ACM, (2001), pp. 327–340.
- [14] H. Huang, T.-N. Fu and C.-F. Li, "Painterly rendering with content-dependent natural paint strokes", Visual Computer, vol. 27, no. 9, (2011), pp. 861–871.
- [15] D. Kang, Y. Ohn, M. Han and K. Yoon, "Generation of coherent mosaic animations: enhancement and evaluation of temporal coherence", Computer Animation Virtual Worlds, vol. 23, no. 3-4, (2012), pp. 191–202.
- [16] H. Kang, S. Lee and C. Chui, "Coherent line drawing", In: Symposium Non-Photorealistic Animation and Rendering (NPAR 2007), (2007), pp. 43–50.
- [17] A. Kasao and K. Miyata, "Algorithmic painter: a non-method to generate various styles of painting", Journal of Visual Computer, vol. 22, no. 1, (2006), pp. 14–27.
- [18] Y. Kim and A. Varshney, "Saliency-guided enhancement for volume visualization", IEEE Transactions on Visualization and Computer Graphics, vol. 12, no. 5, (2006), pp. 925–932.
- [19] J. E. Kyrianiadis, H. Kang and J. D'ollner, "Image and video abstraction by anisotropic kuwahara filtering", Computer Graphics Forum, vol. 28, no.7, (2009), pp. 1955–1963.
- [20] C. H. Lee, A. Varshney and D. W. Jacobs, "Mesh saliency. ACM Transaction on Graphics, vol. 24, no. 3, (2005), pp. 659–666.
- [21] H. Lee, S. Seo, S. Ryoo and K. Yoon, "Directional texture transfer", In: Proceedings of the 8th International Symposium on Non-Photorealistic Animation and Rendering (NPAR 2010), New York, NY, USA, (2010), pp. 43–48.
- [22] L. Liang, C. Liu, Y. Q. Xu, B. Guo and H.-Y. Shum, "Real-time texture synthesis by patch-based sampling", ACM Transactions on Graphics. vol. 20, (2001), pp. 127–150.
- [23] P. Litwinowicz, "Processing images and video for an impressionist", In: Proceedings of SIGGRAPH 1997, (1997), pp. 407–414.
- [24] L. Neumann and A. Neumann, "Color style transfer techniques using hue, lightness and saturation histogram matching", In: Proc. CAe 05, (2005), pp. 111–122.
- [25] T. Nishiyama, H. Ichino, T. Nakamura, M. Kanoh and K. Yamada, "Impressionist painterly style transfer based on texture localization", In: SIGGRAPH Asia 2011 Posters. SA '11, ACM, New York, NY, USA, (2011), pp. 46:1–46:1.
- [26] S. C. Olsen, B. A. Maxwell and B. Gooch, "Interactive vector fields for painterly rendering", In: Proceedings of Graphics Interface 2005, GI '05, (2005), pp. 241–247.
- [27] Y. Park and K. Yoon, "Painterly animation using motion maps", Graphics Models, vol. 70, no. 1-2, (2008), pp. 1–15.
- [28] P. E. Reinhard, "Progressive histogram reshaping for creative color transfer and tone reproduction", In: Symposium Non-Photorealistic Animation and Rendering (NPAR 2010), (2010), pp. 83–90.
- [29] R. M. Haralick and K. S. Dinstein, "Textual features for image classification", IEEE Trans. Syst. Man. Cybern. SMC, no. 3, (1973), pp. 610–621.

- [30] E. Reinhard, M. Ashkhmi and B. Gooch, "Color transfer between images", IEEE Computer Graphics and Applications, vol. 21, no. 4, (2001), pp. 34–41.
- [31] S. Seo and K. Yoon, "Color juxtaposition for pointillism based on an artistic color model and a statistical analysis", Journal of Visual Computer, vol. 26, no. 6, (2010), pp. 421–431.
- [32] A. Soer, "Image categorization using texture features", In: Proceedings of the 4th International Conference on Document Analysis and Recognition. ICDAR '97, (1997), pp. 233–237.
- [33] K. Subr, C. Soler and F. Durand, "Edge-preserving multi-scale image decomposition based on local extrema", In: ACM SIGGRAPH Asia 2009 papers. SIGGRAPH Asia '09, NY, USA, URL <http://doi.acm.org/10.1145/1661412.1618493>, (2009), pp. 147:1–147:9.
- [34] K. Zeng, M. Zhao, C. Xiong and S.-C. Zhu, "From image parsing to painterly rendering", ACM Transaction on Graphics, vol. 29, no.1, (2009), pp. 2:1–2:11.
- [35] M. Zhao and S. C. Zhu, "Sisley the abstract painter", In: Proceedings of the 8th International Symposium on Non-Photorealistic Animation and Rendering. NPAR '10, (2010), pp. 99–107.

Authors



Hochang Lee, He received his B.S., M.S., and PhD degrees from Chung-Ang University, Seoul, Korea, in 2006, 2008, and 2013 respectively. He is currently working in Agency for Defense Development. His research interests are in the area of image processing and non-photorealistic rendering.



Dongwann Kang, He received his PhD. degree in Chung-Ang University (CAU), Korea in 2013. He also received his B.S. and M.S. degrees in Computer Science and Engineering from CAU in 2006 and 2008. He was a research fellow in CAU from Mar. 2013 to Jun. 2015. Now, he is the post-doctoral researcher in the School of Design, Engineering & Computing, Bournemouth University, UK. His research interests include stylization, emotional computing, image manipulation and GPU processing.



Taemin Lee, He received his B.S and M.S. degrees from Chung-Ang University, Seoul, Korea, in 2011 and 2013. He is currently working toward his PhD in Computer Science and Engineering in Chung-Ang University as well. His research interests are in the area of painterly rendering and non-photorealistic rendering.



Kyunghyun Yoon, He received his B.S. degree from Chung-Ang University. He received his M.S. and PhD degrees from the University of Connecticut in 1991. Since 1991, he has been a TENURED ASSOCIATE Professor in the Department of Computer Science & Engineering at Chung-Ang University in Korea. His research interests are in the area of non-photorealistic rendering, color theory.