# Adaptive Brush Stroke Generation for Painterly Rendering

Youngsup Park† and Kyunghyun Yoon‡

Computer Graphics Lab., CS&E, Chung-Ang University, Seoul, Korea

**Abstract**

*We propose adaptive brush stroke generation for source images, using reference data. Colors used are formed by actual palette colors from artists. To create the palette, we have referred mostly to colors used in Van Gogh's works and determined the color of brush strokes by transferring it to the most similar one, through comparing colors used in source images and the palette colors. Also, by referring to the edge orientation of source images, we have applied a brush stroke orientation that surrounds the edges. The sizes were determined depending on the different sizes of the objects from wide to narrow brushes. Finally, we applied spline curve shapes. The brush strokes created in such method, were applied separately according to its segmented images, and composed after rendering.*

## 1. Introduction

Painters all have their own styles that they express though the color, orientation, shape and size of their brush strokes. The color and orientation of the brush strokes are largely dependent on the artist's motives. For example, Van Gogh had his own palette of colors, used curved rather than straight lines, superimposed action on to a calm subject, and expressed the effect of intense, bright colors using the contrast of complementary colors. In this paper, we will analyze the unique characteristics of Van Gogh's works, and use their analysis to propose a method of generating adaptive brush strokes based on the foundation of the analysis.

Most painterly rendering algorithms utilize simple brush strokes of the same shape and size. For this reason, the final image guess a machine-like implement, when compared with an artist's brushwork. We aim to improve computer-based painterly rendering by using Van Gogh's palette colors, brush stroke orientation that follows the edges in the image, various sizes of brush, and long curved strokes.

## 2. Related Work

Litwinowicz[Lit97] pioneered painterly rendering, by creating brush stroke that possesses both line and texture, and a direction determined by the orientation of features in the input image. He also varied the effect by adding random noise to the parameters of each brush strokes, and introduced a brush stroke clipping method that accommodates edges within the image. However, because colors were taken from the input images and the brush strokes were straight, effects conveyed in the original pieces remained exclusive. Hertzmann[Her98] proposed a method, to articulate brush strokes based on the geometry of the input image. This system creates brush strokes that follow spline curves that follow the grain of the image. The picture is built up from many layers, with identical brush strokes in each. A painterly sequence of wide to narrow brushes is used on each layer. However, this method is still unable to express the texture of the brush strokes shown in actual art works, and demonstrates a blurring effect that results from combining brush strokes with different properties. we have also used reference data in determining rendering parameters. Colors are determined by referring to Van Gogh's palette colors using the color transfer method. The orientation of the brush strokes is determined by referring to edge orientation. Brush strokes of diverse sizes are created, and objects are expressed by segmented area clipping rather than by applying a separate clipping method to each brush stroke.

## 3. Adaptive Brush Stroke Generation

The creation of adaptive brush strokes based on reference data requires two steps. First the reference artwork is entered

† aupres98@hanmail.net

‡ khyoon@cglab.cse.cau.ac.kr

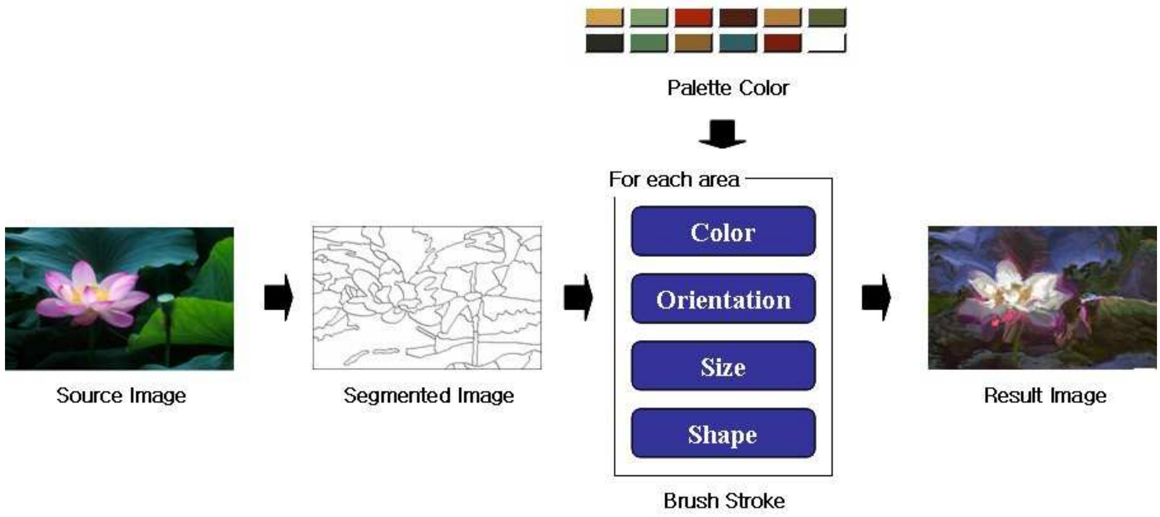



**Figure 1:** *System Flow*

to establish palette colors used. Second, the brush stroke parameters such as color, orientation, size and shape are found. The color of a brush stroke is determined by selecting a pair of similar colors, one from the source images and one from the palette, and then applying the color transfer algorithm. The brush stroke orientation is a spline curve that follows the orientation of image edges within a specified range of a pixel perimeter. The brush size is reduced as rendering progresses. We also employed the Deng and Manjunath method[DB99] to segment the source image. Figure 1 is a flowchart that summarizes our algorithm.

## 3.1. Color

An artist's use of color is largely dependent on their motives and each individual has their own unique palette. We have analyzed the range of colors used by well-known artists(concentrating mainly on Van Gogh, who used a lot of contrast) and we have also generated palette colors based on color theories. By color transfer between a palette reconstructed in this way, and the colors are the input image, we hope to express something of the style of a particular artist.

### 3.1.1. Palette Color Generation

There is no way of knowing the precise palette colors Van Gogh used : we can only atempt to reconstruct them from his work. Our method for doing this can be divided into three steps as shown in Figure 2. First, we input the reference image. Second is process color quantization using Deng et al's method[DKMB99], which preserves image quality while reducing the number of colors. The results of quantization may be assumed to be representative colors and in the final stage of our algorithm, they are added into the palette. If several

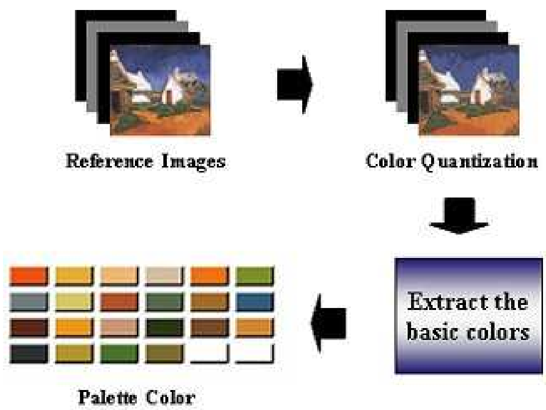



**Figure 2:** *Palette Color Generation*

works are used as references, this may result in similar colors on the palette, which is then regenerated by calculating the average of two colors when the distance between them is lower than a threshold T.

### 3.1.2. Color Transfer

Color transfer, as proposed by Reinhard[DTC98, EMBP01], uses the $\ell\alpha\beta$ color space, in which there is usually little or no correlation between each channel. The $\alpha$ axis represents a chromatic yellow-blue channel, the $\beta$ axis represents a chromatic red-green channel, and the $\ell$ axis represents an achromatic channel. We calculate statistical data for each axis in the $\ell\alpha\beta$ color space, is then used for color transfer between a set of representative colors in the source image and a set of representative colors from the reference images.

Color transfer is a one-to-one pixel matching that uses a linear map to match the means and variances in the $\ell\alpha\beta$ space using Equation ♯1[DTC98, EMBP01]:

$$x_{target} = \frac{\sigma_{reference}}{\sigma_{source}}(x_{source} - m_{source}) + m_{reference}, \quad (1)$$

where $m_{source}$ and $m_{reference}$ are the mean of each axis, and $\sigma_{source}$ and $\sigma_{reference}$ are the standard deviations of each channel.

(a) Source Image (b) Color Transfer Image

**Figure 3:** *The source image and the same image recolored using the color palette Figure6(B)*

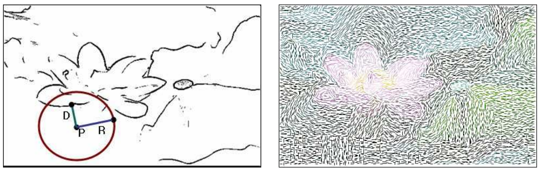
### 3.2. Orientation

Artists customarily draw objects by following their outline. Litwinowicz[Lit97] proposed a local gradient interpolation that determines the orientation of a brush stroke using the gradient of the image, and supplementing that information when the magnitude of the gradient vector is very mall. But it remain difficult to find the orientation of a brush stroke in areas such as a blue sky. To avoid this problem, we interpolate the gradient at a pixel, using distance-weighted edge information from the input image:

$$G'_{(x,y)} = \sum_{i=1}^{n}\left(\frac{G_i}{W_i}\right) \quad (2)$$

$$W_i = \left(\frac{MD}{D_i}\right)^b \quad (3)$$

Equation 2 is the gradient interpolation function that is used to calculate the weighted gradient value at each pixel (x,y). The parameter $G_i$ is the gradient of each pixel $e_i$ and the $W_i$ is the weight of each pixel (x,y). Equation 3 is used to calculate the weight of each pixel $e_i$ with respect to pixel (x,y). The parameter MD is the shortest distance between point P and $e_i$, $D_i$ is the distance between $e_i$ and the center point P, and b is a constant. The pixels $e_i$ comprise the set of edges within a circle of radius R. The range of i is n pixels. Point P and radius R(MD * 1.5∼2.0) are explained in Figure 4(a). Figure 4(b) shows both locally determined and interpolated gradient, and we see that it follows the outline of the objects.



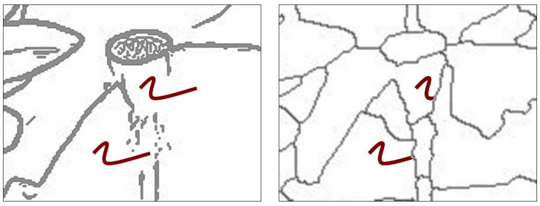

(a) Edge Image (b) Gradient Image

**Figure 4:** *Edge image and gradient image*

### 3.3. Size and Shape

The width of a brush stroke is between 2 and 16 pixels, depending on the size of the segment to which is to be applied. The brush strokes in given area are drawn in order, using smaller and smaller brushes. We used spline curve lines to create curved brush strokes. The control points of the spline curve are chosen to give the appropriate gradient at pixels, and between four nd sixteen control points are generated, depending on the angle at the starting point of a stroke. The width of a spline curve line is created by shifting the spline curve line along an appropriate axis. The shifted spline curves are used to form a polygon, which is colored in, using a color corresponding to the start of the stroke, by a scan-fill algorithm.

### 3.4. Area Clipping



(a) edge clipping (b) area clipping

**Figure 5:** *Edge clipping and area clipping*

In general, artists try to avoid brush strokes that cross in the scene, so that subjects can be distinguished. Litwinowicz[Lit97] applied edge clipping methods in each brush to achieve this effect. However, as shown in Figure 5(a), when the edge information is too detailed, brush strokes become short, and when the edge information not detailed enough edge clipping can not be done accurately. To solve this problem, we consider each segmented area as one layer, and brush strokes are applied separately in each.

## 4. Results

Figures 6 and 7 show two images rendered in the style of Van Gogh using our algorithm. The advantage of using original palettes and long brush strokes typical of this artist are apparent. The color table of the brush stroke in Figure 6 is determined using palette colors from one reference image, while that in Figure 7 is determined using two reference images. Also, the dynamic element that is dependent on the subjective motives of the artist is expressed, as the brush strokes follow the edges.

## 5. Conclusion and Future Work

We have proposed an adaptive brush stroke generation method that produces painterly images. Brush stroke colors are derived from the palettes of well-known artists. Local gradients are supplemented by gradients interpolated from local edges.

The brush strokes in a real painting are determined by palette colors, varying orientation, shape, size, texture, grazing effect, mixed colors and other characteristics. To convey these effects, analyzing and simulating the hydromechanical characteristics of actual paints would be necessary.

## References

[DB99] DENG Y., B.S.MANJUNATH: Color image segmentation. In *Proc. CVPR* (1999), pp. 2446–2451.

[DKMB99] DENG Y., KENNEY C., M.S.MOORE, B.S.MANJUNATH: Peer group filtering and perceptual color image quantization. In *Proc. ISCAS* (1999), pp. 21–24.

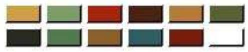
[DTC98] D.L.RUDERMAN, T.W.CRONIN, C.C.CHIAO: Statistics of cone response to natural images : implications for virtual coding. *Journal of Optical Soc. of America 15*, 8 (1998), 2036–2045.

[EMBP01] E.REINHARD, M.ASHIKHMIN, B.GOOCH, P.SHIRLEY: Color transfer between images. *IEEE CG&A 21*, 5 (2001), 34–41.

[Her98] HERTZMANN A.: Painterly rendering with curved brush strokes of multiple sizes. In *Proc. SIGGRAPH '98* (1998), pp. 453–460.

[Lit97] LITWINOWICZ P.: Processing images and video for an impressionist effect. In *Proc. SIGGRAPH '97* (1997), pp. 407–414.

(a) Source Image

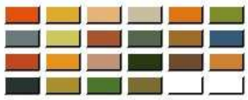


(b) Palette Colors

(c) Result Image

**Figure 6:** *Source and result image using palette colors from a single reference image*

(a) Source Image



(b) Palette Colors

(c) Result Image

**Figure 7:** *Source and result image using palette colors from two reference images*