Real-Time Painterly Abstraction by Pseudo Morphological Filtering

Lei Xiao¹, Hanli Zhao¹*, Xiaogang Jin², and Xiaoyang Mao³

¹Intelligent Information Systems Institute, Wenzhou University, Wenzhou, 325035, China
²State Key Lab of CAD&CG, Zhejiang University, Hangzhou, 310058, China
³University of Yamanashi, Kofu, Japan

We present a simple yet effective non-photorealistic rendering technique that automatically produces painterly abstractions from photographs. We first review the mathematical morphology and related notions, and abstract images using morphological filters. We then design a highly anisotropic morphological filter by integrating with both directions and magnitudes of image gradients. The new filter outperforms many abstraction techniques in terms of the directionality expression and feature enhancement. In addition, the proposed algorithm is simple, parallel, easy to implement, computationally efficient, and fully automatic. A GPU-accelerated implementation is also presented to enable a real-time online rendering. Extensive experiments were conducted to demonstrate that the proposed approach is capable of making various impressive oil-painting-like artistic illustrations.

Keywords: Non-Photorealistic Rendering, Image Abstraction, Real-Time, CUDA.

1. INTRODUCTION

Image based non-photorealistic rendering (NPR) has proliferated over the last few years. A raw photograph often contains more information than necessary and the artist usually uses an abstracted version for efficient visual communication. Image abstraction refers to the task of removing minor visual cues while retaining or even exaggerating important features in the image. It helps convey certain information more effectively as compared with the original photographs. Commercial softwares such as Photoshop support numerous functions for modifying images to produce an artistic effect. The goal of these softwares is to facilitate the expression of artists and also to help novices make interesting illustrations.

Hard and rough boundaries are visible between different brushy strokes in real paintings. Intuitively, texture-based image stylization approaches¹ ² are able to produce high-quality brushy effects by appropriately blending brush textures collected from professional artists. While impressive, the example-based painterly rendering systems involve complex and time-consuming semantic parsing and texture mapping, making them rather tedious for interactive abstraction.

Filter-based approaches are the alternatives to imitate such painterly look. As high-contrast stroke boundaries usually indicate the boundaries of different shapes, most of existing filter-based image abstraction methods use edge-preserving filters instead of mapping of example textures. In particular, the Bilateral filter,³ the mean shift,⁴ and the Kuwahara filter⁵ are the best known ones. With the edge-preserving filtering, low-contrast regions are flattened while high-contrast object boundaries are effectively retained. Stroke orientations in real oil paintings are usually stylized and exaggerated for expressing the imagination of artists. One advantage of filter-based painterly systems is the efficiency. However, these filters fail to stylize the shape boundaries by themselves and require additional processing (e.g., edge detection) to emphasize these boundaries.⁶ ⁷ Rather than composing boundary lines, it is preferred to produce abstracted shape interiors as well as stylistic and rough shape boundaries using a unified image filter.

In this paper, we present a filter-based framework that stylizes both interiors and boundaries of shapes in a brushy-like manner. In order to achieve the artistic oil-painting-like effect, we develop a pseudo morphological filter by integrating with both directions and magnitudes of image gradients. The abstraction process simply includes the smoothing of the gradient-based vector field and the morphological-like filtering along the vector field. As compared with existing abstraction techniques, the new filter improves the abstraction performance in terms of feature enhancement and brushy look (as shown in Fig. 1), which are found in real oil paintings.

The main contribution of this paper is the design of a unified gradient-driven pseudo morphological filter that not only clusters low-contrast regions but also enhances the high-contrast features and, at the same time, produces brushy-like shape boundaries.

*Author to whom correspondence should be addressed.
In addition, the algorithm has many good properties that it is simple, parallel, easy to implement, computationally efficient, and fully automatic. To explore the performance of our algorithm, we have parallelized it using NVIDIA’s CUDA® programming model. Even novices enjoy the process of converting their own photographs into the abstracted version using our developed system.

2. RELATED WORK

Image abstraction and stylization is a topic of increasing interest within the field of computer graphics, and many techniques have been developed to render images or videos for the purposes of both aesthetics and visualization.

There have been a variety of texture-based painterly systems. Early work in this area is that of Litwinowicz,7 who uses image and video analysis to generate impressionistic effects. This system supports oriented brush stroke textures based on smoothed gradient directions. Hays and Essa1 analyze the image into multiple layers to emulate the coarse-to-fine painting process and employ radial basis functions (RBF) to globally interpolate into multiple layers to emulate the coarse-to-fine painting process. Hertzmann12 presents a framework for painting with different brush sizes and curved brush strokes. DeCarlo and Sentella6 employ the mean-shift filter5 for region segmentation of smooth and Canny edges13 for the enhancement of high-contrast shape boundaries. Some researchers,14–15 modify the standard mean shift to solve the video abstraction problem, focusing good spatio-temporal coherence. Winnemöller et al.7 present a semantics-driven approach for stroke-based painterly sys-

tems. Early work in this area is that of Litwinowicz,7 who uses image and video analysis to generate impressionistic effects. This system supports oriented brush stroke textures based on smoothed gradient directions. Hays and Essa1 analyze the image into multiple layers to emulate the coarse-to-fine painting process and employ radial basis functions (RBF) to globally interpolate directions from the strongest gradients. Recently, Zeng et al.2 present a semantics-driven approach for stroke-based painterly rendering using image parsing techniques10 in computer vision. Zhao and Zhu11 introduce an interactive abstract painting system named Sisley based on the psychological principle. High-quality abstracted effects are generated with texture-based painterly systems as the cost of long time semantic parsing and texture mapping.

Instead of texture examples, many researchers exploit edge-preserving filters to fulfill the task of painterly stylization. Hertzmann12 presents a framework for painting with different brush sizes and curved brush strokes. DeCarlo and Sentella6 employ the mean-shift filter5 for region segmentation of smooth and Canny edges13 for the enhancement of high-contrast shape boundaries. Some researchers,14–15 modify the standard mean shift to solve the video abstraction problem, focusing good spatio-temporal coherence. Winnemöller et al.7 use the separate approximation of bilateral filter16 and smooth Difference-of-Gaussian edges to abstract images and videos. Kang et al.17 and Zhao et al.18–19 introduce a feature direction field to improve the feature preservation and stylization. Their direction field is derived automatically from the image gradients and effectively conveys important shapes in the image.20 Some researchers,12–22 demonstrate that the Kuwahara filter3 can be modified to produce edge-preserving painting-like artistic effects from photographs. Although efficient, filter-based painterly systems are difficult to produce stylistic and rough shape boundaries which are visible between strokes in real paintings. In this paper, we analyze the real painting process and propose a filter-based algorithm to emulate the effect of brush strokes.

3. MORPHOLOGICAL FILTERS

In this section, let us review the mathematical morphology which was originally developed for binary images for the analysis and processing of geometrical structures of image objects. It was later extended to grayscale images by using set representations of signals and transforming these input sets via morphological set operations.

All morphological filters are based on two operations: dilation and erosion. Let I be an image and B be a structuring element that contains the relative coordinates of a pixel’s neighborhood. The morphological dilation I ↑ B of I by B at a pixel x and its dual, the morphological erosion I ↓ B, are defined as follows:

\[ (I ↑ B)(x) = \max_{b \in B} \{I(x-b)\} \]

\[ (I ↓ B)(x) = \max_{b \in B} \{I(x+b)\} \quad (1) \]

The dilation enlarges the light features of the image whereas the erosion spreads the dark features. However, due to maximum and minimum noise peaks, there are some light spots and dark spots in Figures 2(a, b). These undesired spots have the shape and size of the structuring element, in our case, a disc with the radius of 5 pixels. The visual artifacts become even worse if the size of structuring element increases.

Both of the dilation and erosion are basic filters combined for complex morphological filters. The morphological opening I ∩ B is then defined as a sequence of one erosion followed by
one dilation \((I \downarrow B) \uparrow B\). The morphological closing \(I \bullet B\) is a sequence of one dilation followed by one erosion \((I \uparrow B) \downarrow B\). From the definitions, we know the opening removes the light features of the image whereas the closing removes dark features (see Fig. 2(c, d)). The combination of opening and closing forms the commonly used open-closing \((I \circ B) \bullet B\). As shown in Figure 2(e), the morphological open-closing effectively smooths small textures and cleans noises while preserving edges and corners.

4. OUR APPROACH

Many painterly artworks have the brushy stroke as the basic primitive for expressing object shapes and the orientations of these strokes are often exaggerated and stylized for efficient visual communication. The standard morphological filters pick colors in the regular disc neighborhood based on a simple color comparison. If one pixel holds its local maximum or minimum intensity, however, the intensity-driven filtering schemes will cluster the neighboring pixels and cause visible disc artifacts. As a consequence, the morphological filters fail to transform images into artistic painterly styles.

In this section, we would like to develop a new type of filter based on the idea of mathematical morphology to fulfill the task of painterly-like abstraction of images without any brush textures. In addition, the GPU-based implementation of our real-time painterly system is presented.

4.1. The Proposed Filter

We begin with the introduction of a highly anisotropic structuring element. In order to emulate the linear property of brush strokes, we redefine the isotropic disc structuring element of the original morphological filters as a curvilinear shape which is decided by a vector field. Rather than resorting to additional vector images, we construct the vector field using image gradients, as we assume the fact that high gradients indicate the boundaries of different shapes. We rotate the gradient field by 90° to obtain an initial vector field (see Fig. 1(b)) and then smooth it using both the magnitude difference and the direction difference as convolution weights. With this Bilateral-like filter applied to the original field, vectors with unimportant directions are smoothly aligned to the salient directions except at sharp corners, as shown in Figure 1(c). Consequently, our anisotropic structuring element is a streamline along the smoothed vector field.

The original morphological filters use local maximum or minimum of colors as the diffusion regulation. In order to produce brushy-like rough styles, we further adopt the local minimum of gradients to guide our filtering. Therefore, the proposed pseudo morphological filter is defined as follows:

\[
F(x, B) = I(x + \arg \min_{b \in B} \|g(x + b)\|) \tag{2}
\]

Where \(g\) denotes the magnitude of the gradient. This equation simply copies the colors with local gradient minima to the output image.

The new filtering is driven by gradient magnitudes as well as salient gradient orientations. Now let us analyze how the proposed simple filter modifies the appearance of the input image. First of all, it effectively flattens colors inside each streamline. Second, if neighboring streamlines are of low contrasts, the whole low-contrast region is even flattened, resulting in painterly-like effects. Third, if there are neighboring streamlines with high contrasts, they probably lie on the boundaries of different brushes and their contrasts will be further increased after filtering. Artists usually choose salient gradient directions as the orientations of brush strokes for exaggeration of object features and our curvilinear structuring element is highly conformable to this case.

4.2. GPU Implementation

In this subsection, we introduce our GPU-based painterly abstraction technique by equipping the proposed pseudo morphological filter. We first convert the input RGB color space (Fig. 1(a)) into the CIE-Lab space which has one intensity channel and two color channels (Fig. 1(d)). Since intensity carries a large portion of structural information in the image, the vector field is then constructed using the intensity image. We can operate on the intensity channel only as suggested by Winnemöller et al. However, as shown in Figures 3(a, b), unwanted color halos occur on high-contrast shape boundaries. The color is green between the trees and the sky in the original image (see Fig. 1(a)) whereas some pixels turn to brown in the intensity only abstraction image.
Iterative control of the level of abstraction: (from left to right and top to bottom) the input image and 1 to 5 passes of the pseudo morphological filtering, respectively.

Fig. 4. Iterative control of the level of abstraction: (from left to right and top to bottom) the input image and 1 to 5 passes of the pseudo morphological filtering, respectively.

This is because high-contrast regions of intensity image are not exactly the boundaries of color shapes. Actually, as shown in Figure 1(c), all the three channels are filtered respectively to avoid such issue. After abstracted, we convert back to RGB space to display the output image (see Fig. 1(f)) and the zoomed view (see Fig. 3(c)).

Note that one pass of pseudo morphological filtering cannot abstract the image too much. Fortunately, the proposed pseudo morphological filtering is iterative as well as incremental. We can intuitively control the level of abstraction with a desired number of filtering passes and just stop the process when the desired level is reached. Empirically, 3 passes are enough to produce stylized brushy-like effects.

The GPU has evolved into a highly parallel, multi-threaded, and many-core processor with tremendous computational horsepower and very high memory bandwidth. Thanks to the powerful GPU, more and more single-threaded CPU-based algorithms become real-time interactive applications in recent years. We implement our GPU-based abstraction framework using NVIDIA CUDA. The construction of vector field is implemented using the separable scheme as suggested by Zhao et al. Given that the kernel widths for the construction of vector field and the pseudo morphological filtering are $m$ and $n$ pixels, respectively. The separable scheme for the Bilateral-like filter of the vector field has linear $2m$ time complexity while the gradient-driven pseudo morphological filter has linear $n$ time complexity.

The proposed framework is rather simple, easy to parallelize, and computationally efficient. We have implemented the proposed image abstraction framework on a PC with a 2.80 GHz Intel Core i5 760 CPU and an NVIDIA GeForce GTX 480 GPU. For a color image with the resolution of $512 \times 512$ and $m = n = 11$, the proposed GPU-optimized algorithm is able to perform the painterly stylization at 35 frames per second. The GPU-accelerated program enables the efficient abstraction of online images.

5. EXPERIMENTAL RESULTS

We demonstrate the control of the level of abstraction with the iterative manner in Figure 4. Given an input image, we can intuitively obtain the different levels of abstraction with the desired passes of filtering. With the increasing number of filtering pass, the brushy effect becomes clearer. Here we provide the corresponding report of the GPU performance in Table I.

Figure 5 shows the comparison of our method with various state-of-the-art image abstraction filters. Figure 5(a) is the input photograph taken from the real world. Figure 5(b) is the result of the standard Bilateral filter, which is used by Winnemöller et al. Due to the isotropic nature, the whole image is flattened except at high-contrast edges with the Bilateral filter. Figure 5(c) is the filtered version of Zhao et al.’s anisotropic Bilateral filter. By smoothing the image along and perpendicular to the flow field, the anisotropic Bilateral filter enhances and stylizes the directional-ality of the image. Figure 5(d) is the abstracted result using the standard Kuwahara filter proposed by Papari et al. Similar to the standard Bilateral filter, the standard Kuwahara filter cannot stylize the boundaries of shapes. Figure 5(e) is obtained after the anisotropic Kuwahara filter by Kyprianidis et al. Note, both the anisotropic Bilateral filter and the anisotropic Kuwahara filter fail to produce brushy-like painterly effects. Figure 5(f) is the result of the proposed filter. The proposed filter improves the abstraction performance in terms of feature enhancement and brushy-like look.

We show various painterly abstraction results in Figure 6. See how the underlying shape features are stylized and how the painterly brushy-like effects are produced. All these vivid pictures demonstrate the feasibility of the proposed algorithm.

Table I. The experimental running performance.

<table>
<thead>
<tr>
<th>Filtering pass</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>FPS</td>
<td>85</td>
<td>45</td>
<td>30</td>
<td>25</td>
<td>20</td>
</tr>
</tbody>
</table>
Fig. 5. Comparison of the proposed filter with various state-of-the-art image abstraction filters.

Fig. 6. Vivid painterly abstraction results.

6. CONCLUSIONS

In this paper, we have presented a new image abstraction framework that stylizes images in a brushy-like painterly manner without any brush textures. A novel pseudo morphological filter is developed by adapting the standard morphological filter with both directions and magnitudes of image gradients. Experimental results and comparisons have demonstrated the feasibility.
Currently, spatio-temporal noises would be enlarged when we simply perform per-frame abstraction to videos. In the near future, we plan to extend the proposed algorithm to videos by taking into account the frame-to-frame coherence. Rough and brushy look can be found in real oil paintings.

Acknowledgments: The authors would like to thank the anonymous reviewers for their helpful and constructive comments and suggestions which definitely improve the paper. This work was supported by the National Natural Science Foundation of China (Grant No. 61100146) and the Zhejiang Provincial Natural Science Foundation of China (Grant No. Y1110004). The research of X. Jin was supported by the NSFC-MSRA Joint Funding (Grant No. 60970159), Zhejiang Provincial Natural Science Foundation of China (Grant No. Z1110154) and the National Natural Science Foundation of China (Grant No. 60933007).

References and Notes

Received: 5 May 2011. Accepted: 15 November 2011.