Analysis of Artistic Edge and Corner Enhancing Smoothing

Anıl Bulut
Hacettepe University
Computer Engineering Department
anil.bulut@hacettepe.edu.tr

Ekrem Candemir
Hacettepe University
Computer Engineering Department
ekrem.candemir@hacettepe.edu.tr

Abstract

We've analyzed a non-linear local operator which is a generalized version of Kuwahara filter and generalized value-and-criteria filter structure(VCFS). VCFS is used to to extended the kinds of operations that can be used within a geometric filter structure[1]. This operation that we analyzed, creates paint-like image effects. In fact, the process is generally an edge preserving smoothing operation. However, this operator gives more successful results when considering restrictions that others have. This operation has some approaches like choosing circular regions and weighting to overcoming restrictions in Kuwahara. As a result of this operation we get more sharper edges and smoothed texture details.

1. Introduction

Low pass filters are used quite frequently for smoothing. But since these filters affect high frequency components, the edges are also blurred as well. To overcome this problem, the idea of edge and corner preserving smoothing was put forward. There are several edge and corner preserving smoothing filters such as bilateral filters[2] and median filters[3]. But it is better to work on Kuwahara filter to get artistic results.

Kuwahara filter is a non-linear filter which basically preserves the edges while image smoothing operation[4]. Its basic working principle is like mean filter. It replaces the current pixel with the mean of neighboring *nxn* block that has least variance. It divides the source image into regions, calculates the variance and mean values of each region, and finally reflects the result to center pixel. In this method, the edges are not affected by the filtering process. So we can protect the position and magnitude value of edges.

But Kuwahara filter has some limitations. Block structure in square shape causes some problems in high texture images. Also there is a bigger problem. If subregions have same minimum standart deviation value, there will be an ambiguity. Algorithm chooses randomly one of this subregions. Some methods, such as taking an average, try to remove these problems, but not exactly the desired results. It also gives unstable results in noisy shadowed areas.

Figure 1: Example of Kuwahara Filter [5]

The filter that we study, is basically an edge and corner preserving filter. It smooths the texture details while preserving edges. However, the operator that we analyzed, is trying to overcome these constraints that Kuwahara has. The most important difference of this method from Kuwahara is; it uses a different method instead of calculating minimum standart deviation directly. It creates weighting windows with Gaussian mask as selection criteria. And local averages and standart deviations are computed as convolutions. Also this method uses circular regions for masking. Circular regions are suitable for preserving the edges and sharp corners. Thus, at the end of this process paint like results are obtained.

2. Related Work

One of the first examples of filter work that protects the edges is the Kuwahara filter, in 1976[4]. Originially this filter was proposed for medical purposes. In the following years, the Kuwahara filter has been used for various purposes and has been extended. Different shaped subregions were created and studied[6]. This was an important step for development in this area.

In subsequent studies, the Kuwahara filter was tried to use Gaussian weighted local average instead of local average to give better results[7]. And there are also some works done in color pictures to solve the problems which caused from different values in each channel[2]. In the following years, some studies were carried out for the purpose of artistic use of this filter.

One of the these studies is[10] which use structural morphological closing. This filter smooths noise and texture details while preserving edges and corners. Even so, there is a problem for this filter that creates many undesirable points. Another studies is morphological area open-closing[11] doesn't create the points and sometimes we get good painting-like images.

Median[3] and bilateral[2] filtering are also good for smoothing textures while preserving edges and corners. The differences between our approach and these filters is these filters don't provide a painting-like images.

3. The Approach

The main idea of paint-like image effect is removing texture details and increasing edge information. We need Gaussian mask for noise rejection. It has also important role in blurring texture details. We create this mask according to given sigma value.

$$g_\sigma(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \qquad (1)$$

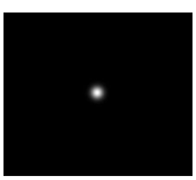


Figure 2: Gaussian Mask

This method works slightly different in color and gray images. If we separate RGB images into the channels and perform this operation to each channel separately, the results could be unsatisfying due to different average and standart deviation values for each channel. To avoid this situation, another method was applied. This method will be explained later stages of this section. After that we convert each channel of image to frequency domain. This helps to make blurring operation faster.

To operate weighting procedure, we divide this Gaussian mask to circular regions. This is one of the most important difference from Kuwahara filter. Each equal sector is expressed as polar coordinates. We use a cutting function for defining sectors. The number of sectors is taken from the user as input. Cutting function is a key factor for weighting procedure. Because we get the weighting functions(w_i) from the multiplication of cutting function(V_i) and Gaussian mask.

$$w_i = g_\sigma . V_i \qquad (2)$$

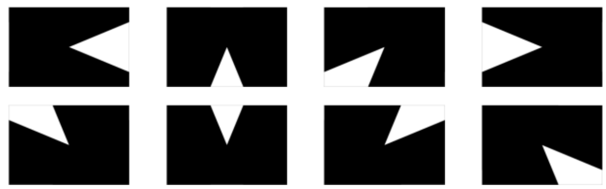


Figure 3: Eight Cutting Functions

To reach this cutting functions, firstly we create sectors and after that we make blurring operation to this sectors.

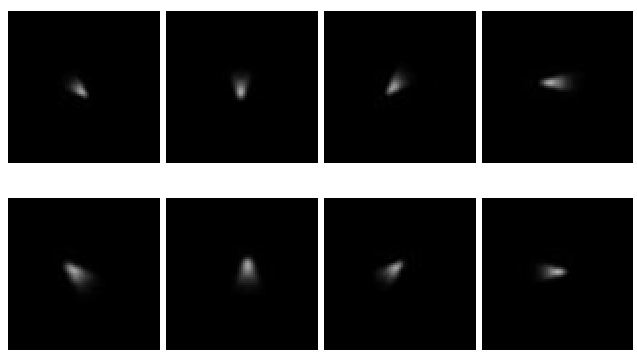


Figure 4: Eight Weighting Functions

This weighting functions have very important role in this algorithm. We can get more satisfying results thanks to this function. Because it affects our selection criteria. After getting this function, we transform each one of them to frequency domain. In the beginning, we transformed the input image to frequency domain. We know that; convolution in spatial domain is equivalent to multiplication in frequency domain[8]. We can get a faster algorithm with multiplication in frequency domain instead of convolution in spatial domain.

After that we calculate local mean(m_i) and standart deviation(s_i) values for each sector and channel according to this formula;

$$m_i = I.w_i \qquad (3)$$

$$s_i^2 = I^2.w_i - m_i^2 \qquad (4)$$

We use a parameter named q, which controls the sharpness of the transition. This parameter has an effect on result image. The effects of this parameter are shown in section Results.

To get sharper edges we should not blur edges with the texture details. Circular regions are suitable for this. Because we can preserve corner and sharp edge information with them. Different sectors are chosen in different image regions. So it blurs some part of images but it preserves the edge area. Finally we get the output image according to this formula;

$$\varphi_q(x, y) = \frac{\sum_i m_i(x, y) s_i(x, y)^{-q}}{\sum_i s_i(x, y)} \qquad (5)$$

The denominator part of formula is used for normalization operation. This formula shows the greatest difference of this filter from the other edge and corner preserving smoothing filters. Result image is dependent on different parameters.

We mentioned that this operator works slightly different for color and gray images. Because in color images, there could be different results for in each channel. To solve this problem we hold three sets of local averages and local standard deviations, for each color channel. And we combined them with;

$$m_i = [m_i^{(1)}, m_i^{(2)}, m_i^{(3)}]^T \qquad (6)$$

$$s_i = \sqrt{\sum_{c=1}^{3} [s_i^{(c)}]^2} \qquad (7)$$

After getting m_i and s_i values like that, we combine them with the same formula. The important point is that; it isn't equivalent to apply the operator to each color component separately.

If image has homogeneus regions, it means s_i values are very similar to each other. In this case filter acts like a standart Gaussian filter. But if image has high edge information in some regions, it means that s_i values are higher than other sectors. And this edges become more sharper with the help of weigthing functions.

3.1. Some Notes and Analyze of Advantages and Disadvantages of This Filter

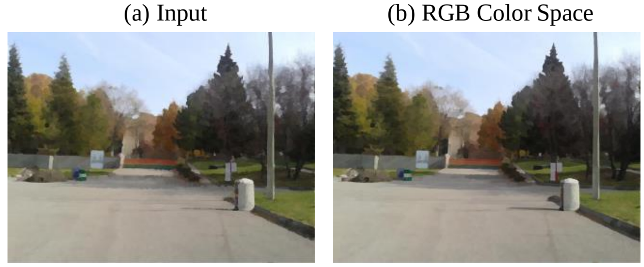
We have said that the biggest difference of this filter from other similar filters is sector selection and weighting functions.

If all sectors have same standart deviation value, in other words if the image contains completely homogeneous regions, filter acts like a Gaussian mask. If the standard deviation of one of the sectors is zero, the output image is entirely dependent on the mean value of that sector. If more than one sector have zero standart deviation value, the output image dependent on the arithmetic mean of these sector's mean value.

When we analyze advantages of this filtering operation; the main advantage is there is no undetermination in case of equal standart deviation values. In this case weighting functions helps us and we can get more satisfying results.
For each subregion, we can say that adaptive filtering is
For each subregion, we can say that adaptive filtering is
done according to its own specifications.

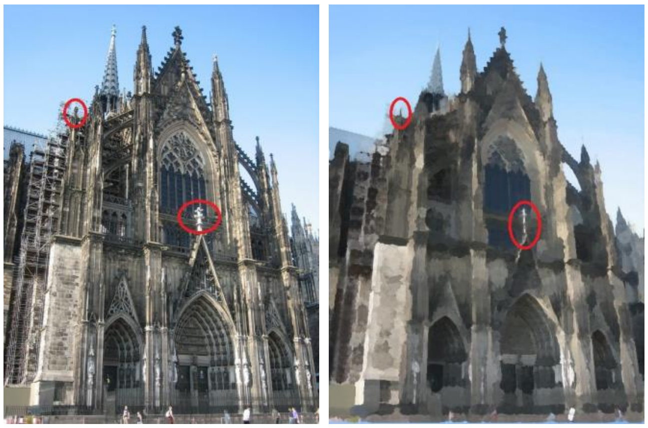
Another advantage is this filter operator gives similar results in different color spaces. It causes from effect of mean and standart deviation value in this filter. Because homogeneity of regions is independent from the color space for standart deviation value. It is also nearly independent for mean value too.

(a) Input (b) RGB Color Space



(c) HSV Color Space (d) L*a*b* Color Space

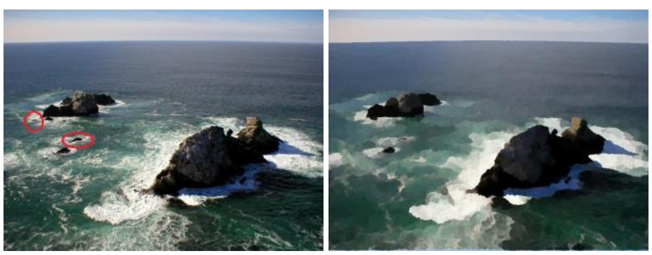
But when we analyze the result image, we can say this as a disadvantage; that the lines in the input image becomes more thinner in some cases.



(a) Source (b) Result

Figure 6

Another disadvantage is that; some small objects are not preserved in some cases.



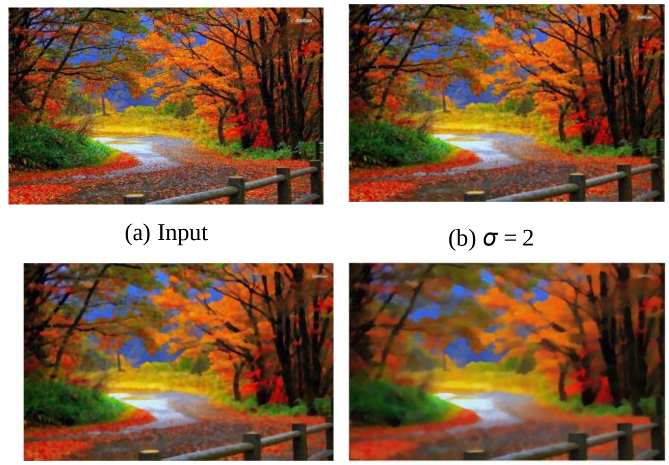
(a) Source (b) Result

Figure 7

4. Effects of Parameters

There are 3 parameters, σ, q and N which effect final images.

We use σ for 2-D Gaussian smoothing kernel which is shown in (1). σ is the standart deviation of the distribution. The degree of smoothing is depended on by standart deviation of σ. If σ is increased, more pixels are smoothed. The effect of increase of the sigma in analyzed filter is like a brush toe. If σ is increased, we can think like brush toe's size increases and brush stroke is getting more thicker, otherwise brush stroke is getting more thinner. You can see σ effect in Figure 8.

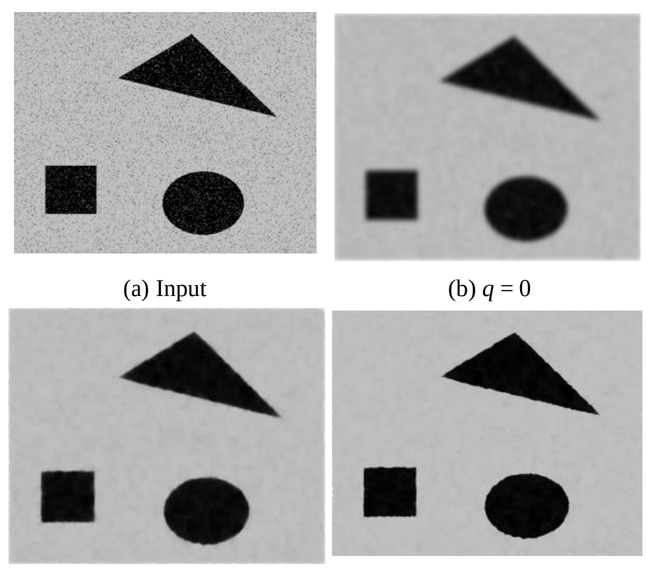


(a) Input (b) $\sigma = 2$

(c) $\sigma = 4$ (d) $\sigma = 8$

Figure 8: Effect of σ

Another parameter q, which controls the sharpness of the transition, is used in (5). We talk about 3 values for q. If $q = 0$, then we get a linear Gaussian filter[Figure 9(b)] . But if q goes to infinity, then we get value and criterion filter. We know that the Gaussian filter is better for noise rejection and value and criterion filter is better for preserving edges[9]. Thus, if q is assigned to a finite numbers, we benefit from the properties of both filters[Figure 9(c)-(d)]. It behaves like a Gaussian filter in homogeneous and textural regions and like value and criteria filter in the edges and in the corners. You can see q effect on Figure 9.

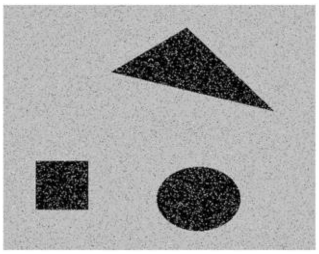


(a) Input (b) $q = 0$

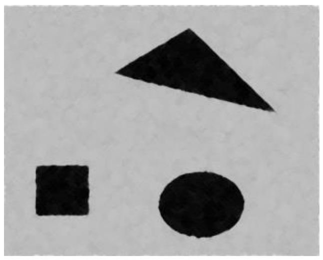
(c) $q = 2$ (d) $q = 8$

Figure 9: Effect of q

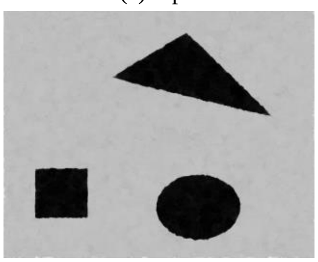
Last parameter N is number of sectors. As the N value increases, we see that the corners are better preserved. It is not possible to tell the effect of N value on the corners, for the edges. As N increases, it is not a significant effect on the edges. You can see differences between edges and corners on Figure 10.



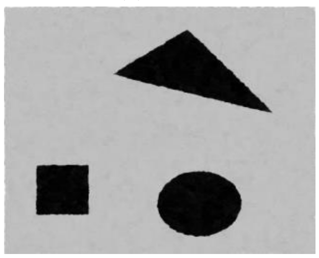
(a) Input



(b) $N = 4$



(c) $N = 8$



(d) $N = 12$

Figure 10: Effect of N

5. Results

(a) Source

(b) Result

Figure 11: Parameters:
σ=4
N=8
q=3.5

(a) Source

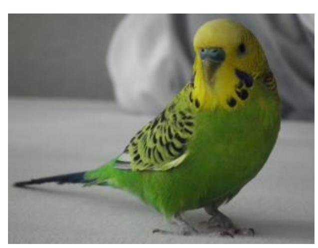
(b) Result

Figure 12: Parameters:
σ=5
N=8
q=5

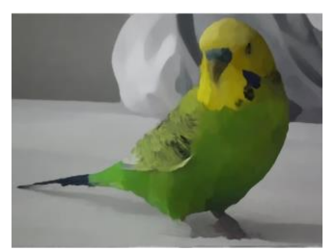
(a) Source

(b) Result

Figure 13: Parameters:
σ=6
N=6
q=6



(a) Source



(b) Result

Figure 14: Parameters:
σ=5.2
N=8
q=6

(a) Source (b) Result

Figure 15: Parameters:
σ=5.5
N=8
q=4.8

6. Conclusion

Basically we've analyzed a edge preserving smoothing filter which is generalized version of Kuwahara filter. In this study, we tried to solve limitations of Kuwahara filter and we obtained painting like results. Edges are sharper like paintings and smooth areas give more blur results.

The two main factors helped us to do this. They were weighting windows and selection criteria. We divide a circular region around each pixel in N equal sectors. This sectors are defined by some cutting functions and Gaussian mask. And we've reached weighting windows using this cutting functions. This operation includes some differences from standart Kuwahara filter.

There will be no undetermination situation in case of equal standart deviation values thanks to weighting windows. However, in some cases, there are disadvantages such as thinning of lines and the inability to protect small objects.

7. References

[1] Mark A. Schulze, An edge-enhancing nonlinear filter for reducing multiplicative noise, Proc. SPIE, San Jose, California, 1997, 46-56

[2] C. Tomasi and R. Manduchi, Bilateral filtering for gray and color images, in Proc. Int. Conf. Comput. Vision, 1998, pp. 839-846

[3] W. K. Pratt, Digital Image Processing. New York: Wiley, 1978

[4] M. Kuwahara, K. Hachimura, S. Ehiu, and M. Kinoshita, Processing of ri-angiocardiographic images, in Digital Processing of Biomedical Images. New York: Plenum, 1976, pp. 187-203.

[5]Kuwahara Filter - Wayne Rasband, 2005 https://imagej.nih.gov/ij/plugins/kuwahara.html

[6] M. Nagao and T. Matsuyama, Edge preserving smoothing, Comput. Graph. Image Process., vol. 9, pp. 394-407, 1979.

[7] R. van den Boomgaard, Decomposition of the Kuwahara-Nagao operator in terms of linear smoothing and morphological sharpening, in Proc. ISMM, pp. 283-292, in 2002

[8] Hacettepe University Department Of Computer Engineering- BBM 413: Fundamentals of Image Processing Lecture Slides http://web.cs.hacettepe.edu.tr/ erkut/bbm413.f16/

[9] M.A. Schulze and J.A. Pearce. "A morphology-based filter structure for edge-enhancing smoothing," 1994 IEEE International Conference on Image Processing (ICIP-94), v. 1, pp. 530-534, Austin, Texas, 1994.

[10] H. J. A. M. Heimans, Connected morphological operators for binary images, Comp. Vis. Image Understand., vol. 73, pp. 99-120, 1999.

[11] A. Meijster and M. H. F. Wilkinson, A comparison of algorithms for connected set openings and closings, IEEE Trans. Pattern Anal. Mach. Intell., vol. 24, no. 4, pp. 484-494, Apr. 2002

- Other Reference: Wikipedia - Kuwahara Filter