Chapter 6

Artistic

Edge and Corner Enhancing Smoothing

Based on the content of the paper

Abstract

Two important visual properties of paintings are the absence of texture details and the increased sharpness of edges as compared to photographic images. Painting-like artistic effects can be achieved from photographic images by filters that smooth out texture details, while preserving or enhancing edges and corners. However, not all edge preserving smoothers are suitable for this purpose. We present a simple nonlinear local operator that generalizes both the well known Kuwahara filter and the more general class of filters known in the literature as «Criterion and value filter structure». This class of operators suffers from intrinsic theoretical limitations which give rise to a dramatic instability due to noise, especially on shadowed areas. Such limitations are discussed in the chapter and overcome by the proposed operator. A large variety of experimental results show that, though no specific painterly style is attempted to be imitated, the output of the proposed operator is visually similar to a painting. Comparisons with existing techniques on a large set of natural images highlight conditions on which traditional edge preserving smoothers fail, whereas our approach produces good results. In particular, unlike many other well established approaches, the proposed operator is robust to degradations of the input image such as blurring and noise contamination.

6.1. Introduction

Smoothing is an important task in image processing [1] – [5]. The best known smoothing technique is low-pass linear filtering. The most widely used filter deploys a Gaussian kernel, since it has been proved to be very close to the optimality for noise rejection [6]. However, since linear low-pass filtering strongly attenuates high frequency components, not only noise but also edges and corners are smoothed out. Therefore, there has been a remarkable effort to find a nonlinear operator able to remove texture and noise, while preserving edges and corners. In the following we refer to such an operator as an Edge and Corner Preserving Smoother (ECPS). Several ECPSs have been proposed in the literature [7]-[12]. The best known ones are based
on median filtering [13], morphological analysis [14], [15], bilateral filtering [16], mean shift [17], total variation [18], and anisotropic diffusion [19]. The latter is probably the most popular ECPS, for which much research has been carried out in the last fifteen years. However, it is not computationally efficient since it requires many iterations to achieve the desired output. A survey of ECPSs can be found in [20].

In the current work, we are interested in a specific aspect of ECPSs: their ability to produce images that are visually similar to paintings. Not all existing ECPSs are suitable for producing such an artistic effect. The class of ECPSs we consider here stems from the early work of Kuwahara [21], where a fast and conceptually simple ECPS is introduced. Specifically, in [21] a symmetric square neighborhood around each pixel of a gray level image is divided in four square windows. The value of the central pixel is replaced by the gray level average over the most homogeneous window, i.e. the window with the lowest standard deviation. Although this operator was not designed for producing artistic images the obtained effects are quite interesting.

The Kuwahara operator has been extended in several ways: by changing the shape of the windows, from squares to pentagons, hexagons and more recently to circles [22], [23], by allowing overlapping windows [24], [25] and by replacing the local averages with Gaussian weighted local averages [25]. In [23], a class of generalized Kuwahara filters, the Value and Criterion Filter Structure is introduced and efficient algorithms in terms of morphological filtering are provided. A formulation in terms of partial differential equations (PDE) is given in [25]. However, all mentioned extensions suffer a common problem: the output of the operator is not uniquely determined when the minimum standard deviation is reached in more than one window. This problem is discussed in depth in this chapter and it is shown that it may have dramatic effects in presence of noise.

In our contribution we propose a new non-iterative ECPS that generalizes the well known Kuwahara filter without suffering the related drawbacks. With respect to the proposed approach, our algorithm makes use of (i) a different set of weighting windows for computing local averages and (ii) a different combination criterion which generalizes the aforementioned minimum standard deviation rule and which does not suffer the above mentioned ill-posedness. The exposition is organized as follows: after a brief review of the Kuwahara filter and a discussion of its limitations in Section 6.2, we introduce the proposed operator in Section 6.3. We present and discuss experimental results in comparison with existing techniques in Section 6.4 and we draw conclusions in Section 6.5.

6.2. Kuwahara filter and extensions

This section gives a short review of the Kuwahara filter, points out its theoretical and practical limitations, and presents an overview of its extensions proposed in the literature.

6.2.1. Review of the Kuwahara Filter

Let us consider a gray level image $I(x,y)$ and a square of length $2a$ centered around a point $(x,y)$ which is partitioned into four identical square subregions $Q_1$, $Q_2$, $Q_3$, and $Q_4$ (Fig. 6.1a):
(6.1)

\[
Q_1(x, y) = [x, x + a] \times [y, y + a]
\]
\[
Q_2(x, y) = [x - a, x] \times [y, y + a]
\]
\[
Q_3(x, y) = [x - a, x] \times [y - a, y]
\]
\[
Q_4(x, y) = [x, x + a] \times [y - a, y]
\]

where the symbol «×» denotes the Cartesian product.

Let \( m_i(x, y) \) and \( s_i(x, y) \) be the local average and the local standard deviation, respectively, computed on each square \( Q_i(x, y), i = 1 \ldots 4 \). For a given point \((x, y)\), the output \( \Phi(x, y) \) of the Kuwahara filter is defined as the local average \( m_i(x, y) \) that corresponds to the \( i \)-th square providing the minimum local standard deviation value \( s_i(x, y) \) [21]. This can be formulated as follows:

\[
\Phi(x, y) = \sum_i m_i(x, y) f_i(x, y)
\]

(6.2)

where

\[
f_i(x, y) = \begin{cases} 1, & \text{if } s_i(x, y) = \min_k \{ s_k(x, y) \} \\ 0, & \text{Otherwise} \end{cases}
\]

(6.3)

Fig. 6.1b shows the behavior of the Kuwahara operator in the proximity of an edge. When the central point \((x, y)\) is on the dark side of the edge (point \(A\)), the output is equal to \( m_4 \) which is the local average computed on the square that completely lies on the dark side (\(Q_4\) in our case). It is the most homogeneous area, having the smallest local standard deviation \( s_i \) among the four subregions. On the other hand, as soon as the point \((x, y)\) moves to the bright side (point \(B\)), the output is determined by another subregion that lies completely in the bright area (\(Q_2\) in our case), since now that subregion has the minimum standard deviation \( s_i \). This flipping mechanism guarantees the preservation of edges and corners, while the local averaging smoothes out texture and noise. In (6.3) it is assumed that only one square can have the minimal standard deviation. The case of more minimal squares is discussed in Section 6.2.2.

6.2.2. Limitations of the Kuwahara Filter

One limitation of the Kuwahara filter is the block structure of the output, particularly evident on textured areas (Fig. 6.2), that is due to the square shape of the regions \(Q_1, Q_4\) and to the Gibbs phenomenon [26]. This problem can be avoided by using different shapes for the regions \(Q\) and by replacing the local averages with weighted local averages. We will discuss this in more detail in Section 6.2.3.
Fig 6.1. Kuwahara filtering: (a) Subregions $Q_i$, $i = 1 \ldots 4$, on which local averages and standard deviations are computed. (b) The subregion with the smallest standard deviation (delineated by a thick line) determines the output of the filter.

Fig. 6.2. (Top) A test image (480x320 pixels), and (bottom) the output of the Kuwahara filter. The block structure due to the Gibbs phenomenon is well visible on the output of the Kuwahara filter, especially on strongly textured areas.
A more serious problem is that the Kuwahara filter is not a mathematically well defined operator: every time the minimum value of \( s_i \) is reached by two or more subregions, the output of the Kuwahara operator cannot be uniquely determined, because it is unclear which subregion should be chosen. As we will illustrate with a simple one-dimensional example, two subregions with equal standard deviations can have considerably different local average values \( m_i \).

For one-dimensional signals, the Kuwahara filter is defined as follows: two local averages \( m_i(x) \) and two local standard deviations \( s_i(x) \) \( (i = 1, 2) \) are computed on the two intervals \([x-a, x]\) and \([x, x+a]\) around each point \( x \). The output at \( x \) is given by the value \( m_i(x) \) which corresponds to the minimum \( s_i(x) \). It is straightforward to see that if we consider the input signal \( I(x) = x \), the two local averages \( m_1(x) \) and \( m_2(x) \) are different, with \( m_2(x) - m_1(x) = a \), while the two standard deviations are equal, \( s_1(x) = s_2(x) = \sqrt{\frac{a}{12}} \). Therefore, for such an input signal, the output of the Kuwahara operator is not uniquely determined. However in real applications where noise is always present it is very unlikely that the estimated standard deviations have the same value, therefore the algorithm will randomly choose either \( m_1 \) or \( m_2 \), depending on the noise. This instability of the output of the Kuwahara operator is illustrated in Fig. 6.3.

![Fig. 6.3 Effect of the indetermination of the Kuwahara operator for a one-dimensional signal. From top to bottom: Input signal \( I(x) = x \) corrupted by very small additive noise, the two local averages \( m_1(x) \) and \( m_2(x) \), the two local standard deviations \( s_1(x) \) and \( s_2(x) \), and the output of the Kuwahara filter. The standard deviations are almost equal and the transitions from \( m_1(x) \) to \( m_2(x) \) and vice versa in the output occur randomly at the zero crossings of \( s_1(x) - s_2(x) \).](image)
Commonly, there are two ways to try to solve this problem [23]: the first solution is to set the output \( \Phi(x) \) equal to \( \frac{m_1(x) + m_2(x)}{2} \) for those values of \( x \) for which the difference \( |s_1(x) - s_2(x)| \) is below a given threshold \( T \). This method is robust to noise but, as pointed out in [23], edges are not very well preserved. Alternatively, if \( |s_1(x) - s_2(x)| < T \), \( \Phi(x) \) can be set to the value of \( m_i(x) \) that is closer to \( I(x) \).

\[
\Phi(x) = m_k(x), \quad k = \arg \min_i \left| m_i(x) - I(x) \right| \tag{6.4}
\]

This latter method results in a better edge enhancement, but it is more sensitive to noise. Moreover, the indetermination discussed above would still be present for an input signal \( I(x) = x \), since \( m_1(x) \) and \( m_2(x) \) are equally close to \( I(x) \).

Fig. 6.4 shows the effect of such a problem for a two-dimensional input signal consisting of a blurred circle corrupted by a very low-power Gaussian noise (Fig. 6.4a). Though the noise in the test image is below the perception threshold, it has a considerable effect on the output of the Kuwahara filter (Fig. 6.4b). On the shadowing corona around the circle, the four values \( s_i \), \( i = 1 \ldots 4 \), are substantially equal to each other and their differences are due only to the noise. Therefore, we obtain an output which varies randomly as the selection jumps from one subregion \( Q_i \) to another, thus producing the instabilities shown in Fig. 6.4b. Fig. 6.4c shows the small improvement brought by the additional criterion (6.4), but the filtered image is still noisier than the input. For comparison, Fig. 6.4d shows the result obtained with the operator that we will introduce in Section 6.3.

### 6.2.3. Generalizations of the Kuwahara filter

In the last two decades, several extensions of the Kuwahara operator have been proposed by changing the number and the shape of the subregions and by replacing local averages with weighted local averages. In [22], the square subregions \( Q_i \) are replaced by pentagons and hexagons. More recently, in [24] circular subregions are considered and in [23], [25] a larger set of overlapping windows is taken into account. In [25], the local averages are replaced by Gaussian weighted local averages in order to avoid the Gibbs phenomenon [26] (Gaussian Kuwahara filtering). All these extensions can be unified in the following general framework: \( N \) weighted local averages \( m_i \) and standard deviations \( s_i \), \( i = 1 \ldots N \), are defined by the following convolutions of the image \( I \) with weighting functions \( w_i \):

\[
m_i = I \ast w_i, \quad s_i = I^2 \ast w_i - m_i^2, \quad \int_{R^2} w_i(x, y) \, dx \, dy = 1 \tag{6.5}
\]

The Kuwahara filter is a special case of this class of filters, with \( w_i(x, y) = 1 \) if \( (x, y) \in Q_i \), and \( w_i(x, y) = 0 \) otherwise, \( i = 1 \ldots 4 \).

For each point \((x, y)\), the output \( \Phi(x, y) \) is obtained by a combination criterion \( C : R^{2N} \mapsto R \):

\[
\Phi(x, y) = C[m_1(x, y), \ldots, m_N(x, y); \ s_1(x, y), \ldots, s_N(x, y)] \tag{6.6}
\]

For both the Kuwahara filter and its extensions, the combination criterion is the *Minimum Standard Deviation Criterion* (MSDC) expressed by (6.2) and (6.3). Therefore, all these methods suffer the indetermination problem discussed in Section 6.2.2.
6.3. Proposed Operator

A computationally appealing algorithm is obtained from (6.5) when all the weighting windows \( w_i \) are translated versions of a given window \( w \). In this case, the filter can be implemented by using only two convolution operations, independently of the number of subregions \( N \) [23]. For this reason, all the possible translated versions of the window \( w \) in a given neighborhood of the central pixels are considered in [23], [25]. This class of filters is called value and criterion filter structure (VCFS) When the used window weighting function \( w \) is Gaussian one gets the so called Gaussian-Kuwahara filter [25].

6.3. Proposed operator

The proposed approach is a generalization of the VCFS. The main contribution of the present study is a new combination criterion that does not suffer the limitations of the MSDC and the use of different weighting functions \( w_i \) which are more suitable for preserving corners and edges. In this section we first introduce the proposed operator for gray level images, and then we present a straightforward extension to color images.

6.3.1. Gray level images.

We divide a circular region around each pixel in \( N \) equal sectors \( S_i \), \( i = 1 \ldots N \), over which we compute weighted local averages and local standard deviations. Let \( g_\sigma(x,y) \) be the following two-dimensional Gaussian kernel:

\[
g_\sigma(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}. \quad (6.7)
\]

The sectors \( S_i \) are defined by the following cutting functions \( V_i(r,\theta) \), expressed in polar coordinates:

\[
V_i = U_i \ast g_{\sigma/4}, \quad U_i(r,\theta) = \begin{cases} N, & i - \frac{1}{2} < \frac{\pi}{N} \theta < i + \frac{1}{2} \\ 0, & \text{otherwise} \end{cases} \quad (6.8)
\]

The weighting functions \( w_i \) are defined as the product between the Gaussian mask \( g_\sigma \) and the cutting functions \( V_i \):

\[
w_i = g_\sigma \ast V_i \quad (6.9)
\]

Fig 6.4. Effect of the indetermination of the Kuwahara filter for a 2D signal: (a) Input image. Output of the Kuwahara filter (b) without and (c) with the additional criterion (4), and (d) output of the operator proposed in this chapter. Unperceivable additive noise in the input image has a strong effect on the output of the Kuwahara filter and its modification according to the additional criterion (4) brings only a small improvement. The approach proposed here is much more stable in this respect.
These weighting functions are a particular case of the theory treated in [27], [28]. Fig. 6.5 shows the weighting functions (6.9) for the case \( N = 8 \) sectors. Note that:

\[
\frac{1}{N} \sum_{i=1}^{N} V_i(x, y) = 1, \quad \frac{1}{N} \sum_{i=1}^{N} w_i(x, y) = g(x, y)
\]  

(6.10)

Once \( m_i \) and \( s_i \) are computed according to (6.5), the mathematical expression of the proposed combination criterion is still the linear combination (6.2) where the discontinuous functions \( f_{i}(x,y) \) (6.3) are replaced by the following functions:

\[
f_{i}^{[q]}(x,y) = \frac{s_{i}^{[q]}(x,y)}{\sum_{i} s_{i}^{[q]}(x,y)}
\]  

(6.11)

where \( q \) is a parameter. Thus, the output \( \Phi_q(x,y) \) is given by:

\[
\Phi_q = \frac{\sum_{i=1}^{N} m_i s_{i}^{[q]}}{\sum_{i} s_{i}^{[q]}}
\]  

(6.12)

where the explicit dependence on \((x,y)\) has been omitted for simplicity of notation. According to (6.12) the filter output \( \Phi_q(x,y) \) is a weighted average of the local averages \( m_i(x,y) \), with weights equal to \( s_{i}^{[q]}(x,y) \). For positive values of \( q \), more importance is given to the values of \( m_i \), that correspond to smaller values of \( s_i \).

For \( q = 0 \), \( \Phi_q(x,y) \) reduces to the arithmetic mean of the values of \( m_i \). Taking into account (6.10), for \( q = 0 \) the proposed operator is equivalent to a linear Gaussian filter. On the opposite extreme, for \( q \to \infty \) the functions \( f_{i}^{[q]}(x,y) \) of (6.11) reduce to the functions \( f_{i}(x,y) \) of (6.3), thus obtaining MSDC.

For finite positive values of \( q \), the proposed operator is an intermediate case between a Gaussian and a value and criterion filter, since it behaves like a Gaussian filter in homogeneous and textured regions, and like a value and criterion filter around edges and corners of objects. Thus, since Gaussian filtering is more suitable for noise rejection and value and criterion filtering is more suitable for preserving edges, we get the advantages of both. The parameter \( q \) controls the sharpness of the transition between the linear Gaussian-like behavior and the VCFS-like behavior.

This behavior is illustrated in Fig. 6.6. On areas that contain no edges (case a), the values \( s_i \), \( i = 1 \ldots N \), are very similar to each other, therefore the output \( \Phi_q \) is close to the average of the \( m_i \) values. The operator behaves very similarly to a Gaussian filter, thus texture and noise are averaged out and the Gibbs phenomenon is avoided. On the other hand, in presence of an edge (case b), the sectors placed across it give higher \( s_i \) values than the other sectors. If \( q \) is sufficiently large (for instance, \( q \geq 4 \)) the sectors intersected by the edge (\( S_5 - S_6 \)) give a negligible contribution to the value of \( \Phi_q \). Similarly, in presence of corners (case c) and sharp corners (case d), only those sectors which are placed inside the corner (\( S_6, S_7 \) for case c and \( S_1 \)}, \( S_2 \) for case d)
for case d) give an appreciable contribution to the value of $\Phi_q$ whereas the other sectors have a negligible effect.

The choice of using circular sector shaped regions $R_i$ as in [28] for computing the local averages is particularly suitable for preserving edges as well as sharp corners. The combination rule expressed by (6.12) allows, for each pixel, to adaptively choose the sectors which provide the best coverage of either the edge or the corner under analysis. Finally, we point out that our combination criterion does not suffer the indetermination of the Kuwahara operator, since no hard selections are involved. Only theoretically, equation (6.12) would be undefined if some $s_i$ is equal to zero, but the indetermination can easily be removed by using the well known De l’Hospital rule. Moreover, since the windows $w_i$ are slightly overlapping, the case $s_i = 0$ can occur only in regions where the input signal $I(x,y)$ is constant.

6.3.2. Color images

The generalization of the discussed method for color images is straightforward: let $f^{(1)}(x,y)$, $f^{(2)}(x,y)$, $f^{(3)}(x,y)$ be the color components in a given color space. The local averages $m^{(c)}(x,y)$ and the local standard deviations $s^{(c)}_i(x,y)$, $c = 1, 2, 3$, $i = 1, \ldots, N$, are computed according to (6.5) for each component $f^{(c)}(x,y)$. Afterwards, the output $\Phi_q$ is computed similarly to (6.12) as follows:

$$\Phi_q = \sum_{i} m_{i} s_{i}^{-q}$$

with

$$m_{i} = \left[ m^{(1)}_i, m^{(2)}_i, m^{(3)}_i \right]^T, \quad s_{i} = \sqrt{\sum_{c=1}^{3} s^{(c)}_i^2}$$

Note that this is not equivalent to applying the operator to each color component independently, since the coefficients $s^{-q}_i/\sum_i s^{-q}_i$ are the same for all three color components.

Fig. 6.5. The eight weighting functions $w_i$ used in the proposed operator.

Fig. 6.6. Sector selection in various situations: (a) homogeneous (texture) area, (b) edge, (c) corner and (d)
sharp corner. The sectors selected to determine the output are delineated by a thick line. If the three components had been treated separately, the coefficients would have been different for each color component and spurious colors would have been introduced on borders of areas with very different chrominance. A similar strategy for avoiding color artifacts has been proposed in [16].

Equation (6.13) can be applied to any color space. We found that the choice of a color space does not significantly affect the final result. In fact from (6.12), we see that the operator output \( \Phi_i \) is determined by (i) the color averages \( m_i \) and (ii) the standard deviations \( s_i \), \( i = 1 \ldots N \). The colors \( m_i \) are independent on the color representation only if the transformation rules between different color spaces are linear. However, for sufficiently homogeneous regions, these differences are not very pronounced also for nonlinear color transformations. Since in (6.13) the major contribution is given by the most homogeneous regions, we conclude that the color averages \( m_i \) are not significantly influenced by a specific choice of a color space, unless images with very low SNR are taken into account.

Formally, the standard deviations \( s_i \), depend on the color space choice as well. However, for natural images, it is very unlikely that the region homogeneity is affected by the employed color space used to represent the image. Therefore in (6.13) a higher weight is given to more homogenous regions independently of the deployed color space. Fig. 6.7 illustrates that the choice of the color space does not affect substantially the results.

6.4. Experimental Results

In this section we present some experimental results illustrating the ability of the proposed operator to generate artistic images. Specifically, In Section 6.4.1 we compare it with other existing ECPS algorithms. The influence of the parameters \( \sigma \), \( q \) and \( N \) is discussed in Section 6.4.2.

![Fig. 6.7. Effect of the different choices of the space color on the smoothing process. From left to right: input image 375x480) and outputs of the proposed operator for the color spaces RGB, YC\(_b\)Cr, L’\(a\)’b’]. The differences between the different outputs are minimal.](image)
6.4.1. Comparison with existing approaches

We compare the proposed approach with Kuwahara [21], Gaussian-Kuwahara [25], median [13], bilateral [16] and morphological [14], [15] filters, which are the most common ECPSs encountered in the literature. As for the Gaussian-Kuwahara filtering, in our work we have implemented a simplified version, by using only four translated Gaussian kernels, centered on the points ($\pm \sigma$, $\pm \sigma$). The images presented here, Figs. 6.8-6.14, are rendered at reduced resolution. Full size images, together with a larger set of examples, are available on the internet at the URL http://www.cs.rug.nl/~imaging/artisticsmoothing Unless differently stated, we use the parameter values according to Table 1.

Figs. 6.8a-b show the outputs of our operator for the test image shown in Fig. 6.2a for two different values of the smoothing parameter $\sigma$ ($\sigma = 1.5$, $\sigma = 3$) defined in (6.7). By comparing them with Fig 6.2b we can see the improvement achieved by our operator with respect to the Kuwahara filter: the blocking effect due to the Gibbs phenomenon is no longer present. The overall effect of the proposed operator is more similar to a painting than the effect obtained by the application of the Kuwahara filter (Fig. 6.2b).

Fig. 6.8c shows the output of a structural morphological closing with a disk as a structuring element. Morphological filtering is able to smooth out texture and noise while preserving edges and corners, but the filtered image presents many undesired spots, due to the noise peaks, having the shape and size of the structuring element.

Morphological area open-closing [15] does not introduce spots and in some cases it can produce nice painting-like images (Fig. 6.8d). However, in several other cases, its output does not substantially differ from the input image, as illustrated by Fig. 6.9. Moreover, due to its flattening property, area open-closing may produce effects that are visually similar to a non-linear distortion of the input image (see, for instance, the change in the brightness of the forehead of the man in Fig. 6.9, third and fourth row, second column with respect to the input image).

Figs. 6.8e and 6.8f show the output of bilateral and median filtering, respectively. Although these techniques have very good performance in smoothing texture while preserving edges and corners, from a perceptual point of view the obtained result is more similar to texture blurring or defocusing rather than to a painting.

In Fig. 6.10 we see the effect of the VCFS ill-posedness on a real image that presents slow and gradual luminance changes in presence of noise (Fig. 6.10a, on the right side of the hurricane). In presence of shadowing, the Kuwahara operator introduces spurious discontinuities (marked by arrows in Fig. 6.10b). These discontinuities are reduced but not completely eliminated by the Gaussian Kuwahara filter (see the arrows in Fig. 6.10c). However, edge preservation is worse, as we can see, for instance, on the right side of the hurricane. In contrast, our operator smoothes out texture and noise, and enhances edges without negatively affecting edges.

Besides preserving edges, the proposed operator sharpens edges that are blurred in the original image. We can observe this effect, for instance, by comparing the right side contour of the hurricane in Figs. 6.10a and 6.10d. This property has a noticeable contribution to achieving a painting effect. It is worth pointing out that, in general, preserving shadowed areas is the hardest task for operators aimed to sharpen blurred edges, since spurious discontinuities tend to be introduced. In this respect, our operator performs well as illustrated by Fig. 6.10d.
The aforementioned edge enhancing property makes our operator robust when acting on blurred images as depicted in Fig. 6.11 where it is shown how the proposed operator performs in comparison with other well established edge preserving smoothers, like the bilateral (Fig. 6.11c) and the median filters (Fig. 6.11d). Specifically, bilateral and median filtering can preserve an edge that is already sharp in the original image but they are not able to sharpen blurred edges. The robustness of our operator to blurring can be exploited in order to obtain large high quality pictures with a painting-like effect when small input images are used, as shown in Fig. 6.12. The test image (445×300 pixels) shown in Fig. 6.12a has been upsampled by a factor of 2 using a standard technique, thus obtaining the blurred image shown in Fig. 6.12b. The output of the proposed operator, shown in Fig. 6.12c, is a sharp-edged high quality image with a painting-like effect. In contrast, the Gaussian Kuwahara operator (Fig. 6.12d) produces a worse result with many discontinuities (like, for instance, the ones on the red grapes or the walnuts).

Another case is presented in Fig. 6.13, where a very small image (102x93 pixels, Fig. 6.13a) has been upsampled by a factor of 6, thus obtaining the blurred and low quality image shown in Fig. 6.13b. Nevertheless, the proposed operator produces a good quality painting-like picture (Fig. 6.13c).

Finally, Fig. 6.14 shows the watercolor effect achieved by the proposed operator with large values of \( \sigma \). Fig. 6.14b shows the output of our operator with \( \sigma = 12 \) for the test image of Fig. 6.12a. Such an effect cannot be achieved by other existing edge-preserving smoothers, like, for instance, the Gaussian Kuwahara filter with the same value of \( \sigma \) (Fig. 6.12c).

### 6.4.2. Influence of the parameters

We now discuss the influence of the values of the parameters \( \sigma, q \) and \( N \) on the performance of the proposed operator. The value of \( \sigma \) controls the size of the stroke used for the painting: as \( \sigma \) increases, the brush touches become coarser (Fig. 6.15). Concerning the number of sectors \( N \), Fig. 6.16 shows the output of the proposed operator applied to a synthetic noisy image (Fig. 6.16a), for \( N = 4, 6, 8 \) (Figs. 6.16b-d). We can see that, for these values of \( N \), the edge preservation is not significantly influenced by the number of sectors. On the other hand, when higher values of \( N \) are used the triangle’s corners, especially the sharper ones, are better preserved, because narrower sectors better fit sharp corners (see Fig. 6.6b). Noise rejection is not significantly influenced by the value of \( N \). In our experiments with a large number of images we found that \( N = 8 \) gives satisfactory results in terms of visual similarity between the resulting outputs and true paintings.

<table>
<thead>
<tr>
<th>Method</th>
<th>Values of the input parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposed approach</td>
<td>( \sigma = 3, N = 8, q = 3 )</td>
</tr>
<tr>
<td>Kuwahara [21]</td>
<td>( a = 5 )</td>
</tr>
<tr>
<td>Gauss Kuwahara [25]</td>
<td>( \sigma = 3 ), four Gaussian windows centered in ( \pm \sigma ), ( \pm \sigma )</td>
</tr>
<tr>
<td>Structural closing [14]</td>
<td>Disk of radius 5 as structuring element</td>
</tr>
<tr>
<td>Area open-closing [15]</td>
<td>( A_{iso} = 50 ) pixel</td>
</tr>
<tr>
<td>Median [13]</td>
<td>( 5 \times 5 ) square window</td>
</tr>
<tr>
<td>Bilateral [16]</td>
<td>( \sigma = 3, \sigma_r = 0.3 ), Each color component ranges between 0 and 1</td>
</tr>
</tbody>
</table>
Fig. 6.17 shows the output of the proposed operator, applied on the test image of Fig. 6.16a, for different values of \( q (q = 0, 2, 8, \infty) \). For \( q = 0 \), all the coefficients \( f_i \), \( i = 1 \ldots N \), are equal (see (6.11)), so that the proposed operator reduces to a Gaussian smoother. As we see in Fig. 6.17a, noise is smoothed, but edges and corners are blurred. As \( q \) increases, edges and corner are better preserved (Figs. 6.17b-d). However, beyond a certain value of \( q \), we observe that there is no further improvement in the edge preservation. This can be easily explained as follows: The case \( q \to \infty \) corresponds to the standard minimum variance criterion, thus noise is averaged only over one sector. Conversely, the case \( q = 0 \) corresponds to the Gaussian smoother and noise is averaged over an area that is \( N \) times larger than in the case \( q \to \infty \). Consequently, for \( q = 0 \) noise rejection is \( N \) times more efficient than for \( q \to \infty \).

Fig. 6.17 shows the output of the proposed operator, applied on the test image of Fig. 6.16a, for different values of \( q (q = 0, 2, 8, \infty) \). For \( q = 0 \), all the coefficients \( f_i \), \( i = 1 \ldots N \), are equal (see (6.11)), so that the proposed operator reduces to a Gaussian smoother. As we see in Fig. 6.17a, noise is smoothed, but edges and corners are blurred. As \( q \) increases, edges and corner are better preserved (Figs. 6.17b-d). However, beyond a certain value of \( q \), we observe that there is no further improvement in the edge preservation. This can be easily explained as follows: The case \( q \to \infty \) corresponds to the standard minimum variance criterion, thus noise is averaged only over one sector. Conversely, the case \( q = 0 \) corresponds to the Gaussian smoother and noise is averaged over an area that is \( N \) times larger than in the case \( q \to \infty \). Consequently, for \( q = 0 \) noise rejection is \( N \) times more efficient than for \( q \to \infty \).

Fig. 6.8. (a-b) Output of the proposed operator for the test image shown in Fig. 6.2a, with (a) \( \sigma = 1.5 \), (b) \( \sigma = 3 \). Results of (c) morphological structural closing [14], (d) morphological area open-closing, (e) median [13] and (f) bilateral filtering [16].
Fig. 6.9. Comparison of the proposed operator with morphological area open-closing [15] First row: input images (555x347 pixels), second row: output of the proposed operator, third and fourth row: output of area opening with $A_{min} = 50$ and $A_{min} = 100$. In these examples area opening does not produce artistic effects.
Fig. 6.10. Example of the effect of the indetermination of the Kuwahara filter on natural images: (a) Test image (160×270 pixels), and the outputs of (b) the Kuwahara, (c) the Gaussian Kuwahara, and (d) the proposed filter. The arrows point at some of the discontinuities introduced by the Kuwahara-like operators. Such discontinuities are not present with the proposed approach.

Fig. 6.11. Comparison of different methods for an image that contains blurred regions: (a) test image (508×409 pixels), and outputs of (b) the proposed operator, (c) bilateral [16] operator, and (d) median [13] filter. The proposed method succeeds in restoring the blurred edges, while the other two fail.
Fig. 6.12. Example of enlargement: (a) original image (222×150 pixels), (b) image enlarged by a factor 2 and outputs of (c) our and (d) the Gaussian Kuwahara operators with the same value of $\sigma = 3$.

Fig. 6.13. Another example of enlargement: (a) Original image (102×93 pixels), (b) upsampled image by a factor of 6 and (c) the output of the proposed operator. The painting-like effect in the output does not suffer the blur of the enlargement.
6.4. Experimental Results

**Fig. 6.14.** Example of the watercolor effect achieved by our operator with a large value of $\sigma$: (a) test image (369×568 pixels), and the outputs of (b) our filter and (c) the Gaussian Kuwahara filter for $\sigma = 12$.

**Fig. 6.15.** Influence of $\sigma$ on the output of the proposed operator. (a) Test image (830×623 pixels) and (b,c,d) output of the proposed operator for $\sigma = 2, 4, 8$ respectively.
Fig. 6.16 Influence of the number of sectors $N$ on the output of the proposed operator. (a) Noisy synthetic test image ($770 \times 340$ pixels), and output of the proposed operator for (b) $N = 4$, (c) $N = 6$, and (d) $N = 8$. The values of the other parameters are $\sigma = 4$ and $q = 8$.

Fig. 6.17 Output of the proposed operator, applied on the test image of Fig. 6.15a, for: (a) $q = 0$, (b) $q = 2$, (c) $q = 8$, and (d) $q \rightarrow \infty$. Noise reduction is better visible in the electronic version of the chapter.

Fig. 6.18 plots the noise reduction ratio $R$ as a function of $q$. $R$ is defined as the ratio between the noise levels after and before the filtering. As we see, $R$ remains rather constant for small values of $q$, and increases substantially for larger values. We observe that the cases $q = 8$ (Fig. 6.17c) and $q \rightarrow \infty$ (Fig. 6.17d) give rise to the same performance in terms of edge and corner preservation, but there are about 4.5 dB of difference in terms of noise rejection in favor of the case $q = 8$. In our experiments with natural images we found that $q = 8$ is a good compromise between noise and texture reduction and edge preservation.
6.5. Discussion and Conclusion

Several ECPSs are commonly deployed in image processing applications in order to smooth out noise and texture while preserving or enhancing edges and corners. The ability to sharpen blurred edges, rather than only preserve the already sharp edges in the original image, gives a noticeable contribution to achieving a painting-like effect in the output images. In this chapter we demonstrated that some very common ECPSs, like bilateral filtering [16] or median filtering [13], are not suitable to accomplish this task, since they are not able to sharpen blurred edges. Moreover, in some cases the resulting smoothing effect looks more like diffusion or blurring of the texture, rather than a painting-like effect (Figs. 6.8e-f). We showed that structural morphological opening and closing, well known ECPSs, do also not give appreciable results since the output image presents spots with approximately the same shape and size as the one of the used structuring element. More sophisticated morphological filters based on area open-closing do not suffer from the mentioned spot problem and in some cases they are able to produce painting-like images. We, however, demonstrated that such methods are not robust to blurring and in some cases they do not change appreciably the input image.

A more suitable class of ECPSs for artistic imaging is the Value and Criterion Filter Structure [23] that stems from the early work of Kuwahara [21] and its subsequent generalizations. These nonlinear filters are able to smooth out texture while enhancing, not only preserving, edges and corners, and they produce images quite similar to paintings. However, we showed in this chapter that such filters are not mathematically well defined operators and give rise to the instability in presence of shadowed areas.

In this contribution we proposed a new EPCS that does not suffer the mentioned limitations. The proposed operator is based on similar principles as the Kuwahara filter but it is mathematically well defined. The chosen form of the weighting functions $w_j$ is particularly suitable for preserving edges and sharp corners and, at the same time, for achieving the same texture rejection level as the Gaussian filtering on edgeless areas.

It is worth pointing out that that our operator, introduced in (6.12), has some similarity with

Fig. 6.18. Noise rejection ratio of the proposed operator, as a function of $q$. 

6.5. Discussion and Conclusions
bilateral filtering [16], defined as:

$$
\Phi_{bl}(r) = \frac{\sum I(r+x) \cdot w(x) \cdot d[I(r), I(r+x)]}{\sum w(x) \cdot d[I(r), I(r+x)]}
$$

(6.15)

In [16], \(w(p)\) is the Gaussian kernel (6.7) and the range distance \(d(x, y)\) is proportional to \(e^{-|x-y|^2/2\sigma^2}\), where \(\sigma\) is an input parameter. Thus, in the weighted average (6.14), the major contribution is given by those pixels which are spatially closer to point \(r\) and whose gray levels are closer to \(I(r)\). The proposed operator has a similar structure: in (6.12), the weighted sum is over the regions instead over the pixels, and the weighting coefficients \(s_{ij}\) are determined by the degree of homogeneity of each region, instead of the range distance between each gray level and the central value \(I(r)\). The spatial term \(w(p)\) does not appear explicitly in (6.12), but is taken into account by the convolutions (6.5).

The novelty of bilateral filtering was that the local average (6.15) is mainly determined by those pixels for which the range distance \(d\) is sufficiently high. This fact makes the filter an ECPS. However, its main limitation is that the pixels for which the term \(d[I(r), I(r+p)]\) is high are spatially unrelated to each other. The factor \(w(p)\) only limits the average (6.15) on a local neighborhood of \(r\). The proposed approach overcomes this limitation by averaging over regions instead over pixels.

From the computational point of view, the proposed operator is more demanding than the VCFS because the weighting windows \(w_i, i = 1 \ldots N\), are not shifted versions of the same window \(w\). Therefore, the number of required convolutions is \(2N\) instead of 2.

To summarize, the most significant differences of our approach to previous approaches are: (i) deploying the weighting windows proposed in [27], [28] for computing local averages and local standard deviations, (ii) a new combination rule which overcomes the ill-posedness of the Kuwahara filter, and (iii) adaptive and automatic selection of the sectors that give a non-negligible contribution to the weighted average (6.12), according to the local pattern around each pixel. Regarding the latter point, all \(N\) sectors will be used in a homogeneous texture, while a single sector may be needed at a corner. One limitation of the method described in this chapter is the inability to produce elongated brush strokes, so that the user can control their size but not their orientation. In the next chapter, the algorithm proposed here will be extended to elongated brush strokes.

Our experiments, performed on a wide set of natural images, show the effectiveness of the proposed approach in comparison with the above mentioned existing ECPS. We demonstrated that, though no specific painterly style is attempted to be imitated, the proposed operator produces a very interesting painting-like effect and that it is robust to image degradation like blurring. We also showed the ability of the proposed operator to produce large high quality painting-like images starting from small input images.

**Bibliography**


6.6. References


6. Artistic Edge and Corner Enhancing Smoothing


