Edge Preserving Filters on Color Images

Vinh Hong[1], Henryk Palus[2], and Dietrich Paulus[1]

[1] Institut für Computervisualistik, Universität Koblenz-Landau, Universitätsstr. 1,
56070 KOBLENZ – Germany, {hong,paulus}@uni-koblenz.de
[2] Institute of Automatic Control, Silesian University of Technology, Akademicka 16,
44-100 GLIWICE – Poland, hpalus@polsl.gliwice.pl

Abstract. In this contribution we present experiments on color image enhancement for several different non-linear filters which originally were defined for gray-level images. We disturb sample images by different types of noise and measure performance of the filters. We provide signal-to-noise measurements as well as perceived color difference in ΔE as defined by the CIE. All images and test programs are provided online on the internet so that experiments can be validated by arbitrary users on any image data.

Keywords: color image enhancement, edge-preserving filters, ΔE, performance measures.

1 Introduction

Color images as processed in various applications are recorded by different acquisition devices. Cameras as well as scanners have their specific noise characteristics. Image transmission may as well introduce noise into the image data. Typical models for noise are either Gaussian or salt-and-pepper noise; Gaussian noise is used as a model for sensor errors, drop-outs during transmission and errors on the CCD chip can be modelled by salt-and-pepper noise.

In our contribution we use images from a data base [3] and disturb them by artificial noise of varying degree and type. In Sect. 2.1 we describe some non-linear smoothing filters, such as edge preserving smoothing algorithm [8], and extend them to color images where they were defined for gray-level images originally. We apply these filters to the disturbed input images and compare each result with its corresponding original image to compute difference measures. Standard measures are the signal-to-noise ratio (SNR) and maximum differences for color vectors. As all disturbances are modelled in RGB, we compute these measures in RGB as well. More important for human perception than SNR is the so-called ΔE difference [12] which describes the perceived color difference (Sect. 3). In Sect. 4 we conclude our contribution with a summary of the evaluation and the prospective work.

2   Color Image Processing

Color image processing has become a central part of automatic image analysis as color can provide valuable cues for identification and localization of objects [11].

2.1   Color Filters

In research and literature there exist several filters that can be classified into linear and non-linear filters. Filters can either operate in the spatial or in the frequency domain [5]. In the following we compare non-linear filters in the spatial domain and additionally use an AMF (arithmetic mean filter).

Linear Filters. A huge number of filters for single band images has been proposed in the long history of image processing. As color images became affordable with respect to sensors, memory and processor speed, some of these filters have been extended to color.

If a linear filter, such as a Gaussian or mean filter, is applied to each channel of an RGB image separately, the resulting image will contain usually color triplets which are not present in the input image. Such artifacts yield perceptional differences which can be avoided by non-linear filtering. On the other hand, additive noise, such as additive Gaussian noise, can be removed by a low-pass filter which averages color vectors.

Non-linear Filters. Filters which are quoted to smooth homogeneous areas while preserving edges are the

- EPS (Edge preserving smoothing), presented for gray-level images in [8],
- SNN (Symmetric Nearest Neighbour Filter) described in [9,6],
- K-N (Kuwahara-Nagao Filter) proposed in [7,8] for gray-level images,
- VMF (Vector Median Filter, also known as CVF (Color Vector Median)), presented in [10,1].

EPS Filter. In the so-called *edge preserving smoothing* algorithm introduced in [8], the selection of gray-level pixels for averaging is done based on statistical principles. The algorithm uses nine different 5×5 masks for each pixel; three of them are shown in Fig. 1. The pixels marked in the neighborhood are used for the following computations. The symmetrical use of 1 (a) and (b) results in eight different masks. Each of these masks includes seven points for the calculation of the new gray-level. The contrast mask (c) includes nine elements for the following computations. For each mask we compute the variance. The mask with the lowest variance is selected. The central pixel gets the mean value of all points marked in this mask.

To extend this algorithm to color, we compute the color covariance matrix inside each mask. We decide for that mask for which the Frobenius norm of the covariance matrix is minimal and compute the mean color vector for the resulting pixel. This may, of course, introduce artifacts.
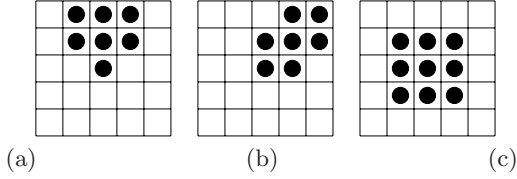
Fig. 1. Masks for edge preserving smoothing

SNN Filter. The SNN is a filter related to the mean and median filters but with better edge-preserving properties. The neighbours of the central pixel in a window are considered as four pairs of symmetric pixels (N-S, W-E, NW-SE and NE-SW). For each pair the pixel closest in color to the central pixel is selected. The colors of these four selected pixels are averaged and the mean color value is a new color for central pixel. The mask for the SNN filter is shown in Fig. 2.
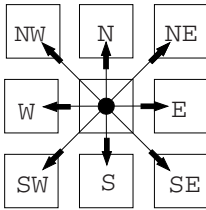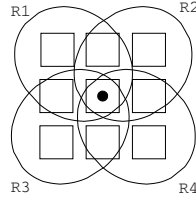

Fig. 2. Mask for SNN filter

Fig. 3. Mask for Kuwahara-Nagao filter

K-N Filter. The 3×3 mask shown in Fig. 3 is split into four 2×2 slightly overlapping windows with the mask's central pixel as a common part. For each window in a gray-level image, the variance is calculated. The mean value of the window with minimal variance (maximal homogeneous region) is used as the output value of the central pixel.

As in the case of the edge-preserving smoothing, we extend this filter to color as we compute the color covariance matrix and use the Frobenius norm.

Recently the gray-level version of K-N filter has been generalized for round windows [2] and it has been shown that the filter is composition of linear diffusion and morphological sharpening [14].

Vector Median Filter. The definition of the vector median of a set of color vectors $\boldsymbol{f}_i$ in a window $\mathcal{W}$ is given in [10] as

$$\boldsymbol{f}_{\mathrm{v}} = \mathrm{argmin}_{\boldsymbol{f}_i \in \mathcal{W}} \sum_{\boldsymbol{f}_j \in \mathcal{W}} ||\boldsymbol{f}_i - \boldsymbol{f}_j|| \quad . \tag{1}$$

In our experiments we use $\boldsymbol{f}_1, \ldots, \boldsymbol{f}_9$ in a square 3×3 window and apply the Euclidean norm on the difference color vector in (1). As this filter does not include averaging, it is the only filter described here which does not introduce color artifacts.

3   Experiments

Signal-to-noise ratio A measure for the accuracy of a filter is given by the *signal-to-noise ratio* (SNR). For color images pairs we define the SNR by a quotient of means:

$$\text{SNR} = 10 \log_{10} \frac{E[\boldsymbol{f}_i{}^{\mathrm{T}} \boldsymbol{f}_i]}{E[\boldsymbol{n}_i{}^{\mathrm{T}} \boldsymbol{n}_i]} \quad , \tag{2}$$

where $\boldsymbol{f}_i$ is the color vector and $\boldsymbol{n}_i$ is the noise vector computed by the vector difference of the two pixels.

Color Metric To measure perceptual color distances between two color stimuli several metrics such in color spaces such as CIELUV, CIELAB, CIE94, etc. can be used [13,15]. In this paper we will prefer the CIE-recommended color metric CIE94 (see equation (6) below). That is a modification of the CIELAB color difference formula (3):

$$\Delta \mathrm{E}_{ab}^* = \sqrt{(\Delta \mathrm{L}_{ab}^*)^2 + (\Delta \mathrm{a}_{ab}^*)^2 + (\Delta \mathrm{b}_{ab}^*)^2} \quad . \tag{3}$$

A value of $\Delta \mathrm{E}_{ab}^* = 1$ resp. $\Delta \mathrm{E}_{CH}^* = 1$ corresponds to the human's eye minimal perceivable difference between two colors. The greater the color difference between two stimuli is, the greater is its $\Delta \mathrm{E}^*$ value [4]. The CIELAB color metric from the CIE (International Commission on Illumination) describes the color difference between two color points in the uniform L*a*b* space. The axes of this color space compound of the lightness-axis L*, the red-green-axis a* and the yellow-blue-axis b* [13]. In this color space the Euclidean distance between two points corresponds to perceived difference[16]. The symbols $\Delta \mathrm{L}_{ab}^*$, $\Delta \mathrm{a}_{ab}^*$ and $\Delta \mathrm{b}_{ab}^*$ represent the componentwise differences (lightness, the red-green and the yellow-blue) between the two colors.

The color difference (3): has several drawbacks in practice [16]. So the CIE introduced an improved color metric called CIE94 which computes the weighted Euclidian distance between two points in the uniform L*C*H* space. This color space uses the polar coordinates chroma C* and hue H* rather than the cartesian coordinates a* and b* of the L*a*b* space. In comparison to the L*a*b* space the L*C*H* space is a more intuitive representation of color, because for example hue can be uniquely computed [16]. Chroma can be computed by

$$\mathrm{C}_{ab}^* = \sqrt{(a^*)^2 + (b^*)^2} \quad , \tag{4}$$

and hue can be caculated from

$$\mathrm{H}_{ab}^* = \arctan \frac{b^*}{a^*} \quad . \tag{5}$$

CIE94 computes the color difference between two colors in the $L^*C^*H^*$ space by

$$\Delta E^*_{CH} = \sqrt{\left(\frac{\Delta L^*_{ab}}{k_L S_L}\right)^2 + \left(\frac{\Delta C^*_{ab}}{k_C S_C}\right)^2 + \left(\frac{\Delta H^*_{ab}}{k_H S_H}\right)^2} \quad . \tag{6}$$

The symbols ΔL^*_{ab}, ΔC^*_{ab} and ΔH^*_{ab} represent the differences between the two given colors corresponding to those lightness, chroma and hue attributes. S_L, S_C and S_H represent parameters calculated from the chroma coordinates of the two color stimuli. k_L, k_S and k_H are parameters those take specific experimental conditions into account[16]

We use the following parameter configuration [16]:

$$k_L = k_S = k_H = S_L = 1 \tag{7}$$
$$S_C = 1 + 0.045\mu C^*_{ab} \tag{8}$$
$$S_H = 1 + 0.015\mu C^*_{ab} \tag{9}$$

The values S_C and S_H are computed from the mean chroma value μC^*_{ab} of the two given color stimuli.

Noise model An ideal color image $\boldsymbol{f}$ consisting of color vectors $\boldsymbol{f}_i$ is disturbed by additive noise $\boldsymbol{\beta}_i$ and multiplicative noise $\boldsymbol{\gamma}_i$

$$\boldsymbol{g}_i = \boldsymbol{\gamma}_i \cdot \boldsymbol{f}_i + \boldsymbol{\beta}_i \tag{10}$$

to yield the observed image $\boldsymbol{g}$. We added zero-mean Gaussian noise $\boldsymbol{\beta}$ with varying σ to images in a test data base where the noise was statistically independent for the color channels. In another experiment we introduced impulsive noise which can be considered as a multiplicative noise $\boldsymbol{\gamma}$ with $\gamma_i = 0$ for drop-outs, $\gamma_i = 1$ for undisturbed image information, and $\gamma_i = 255$ to introduce white spots; with a given probability p white and black spots are created, each with probability 0.5. All test images can be found in a public image data base.[1] For each corrupted image we applied the filters described in Sect. 2.1. An example is shown in Fig. 4.

We then compared original and filtered image and computed SNR and mean ΔE^*_{CH}. Of course, if little noise is added to the image, the values for filtered images are worse than for the unfiltered noisy image, as can be seen from Fig. 5 and Fig. 6. The higher the corruption is, the higher the improvement can be by filtering.

4   Conclusion. Prospective Work

The vector median filter outperforms the other filter methods for impulsive noise, if we use ΔE^*_{CH} as a measure. This is as expected, as a measure for perceived color differences should be sensitive to color artifacts. The vector median filter
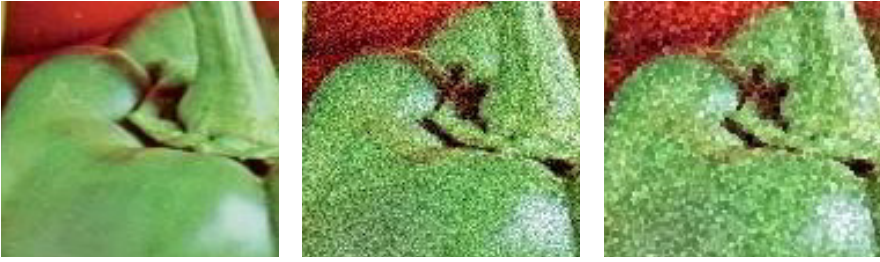
---

[1] http://www.uni-koblenz.de/~puma

Fig. 4. Example image "`peppers`" (left), corrupted image (center), filtered image (right)
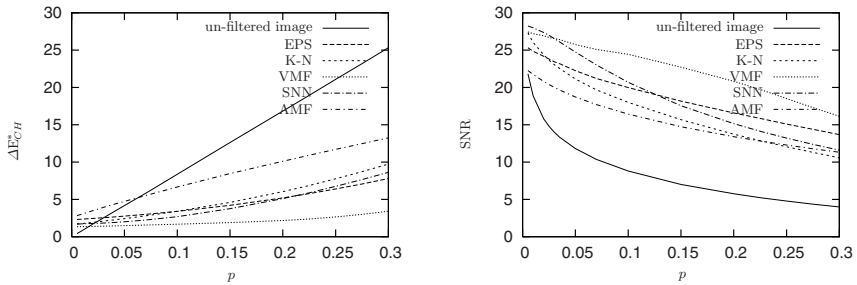

Fig. 5. Example image "`peppers`" corrupted by impulsive noise

also outperforms the other filters in the case of the SNR-measure, if the SNR of the input image is low.

Naturally, linear filtering reduces Gaussian noise better than rank-order filters. The Arithmetic Mean filter returns the best ΔE_{CH}^* for Gaussian noise. In general, the Vector Median filter outperforms the other filters for both distance measures.

Both measures SNR and ΔE_{CH}^* are consistent for extreme cases, i.e. very little or very large noise as they mostly have the same ordering for a qualitative judgement of the filters. Details in the medium range noise reveal differences of
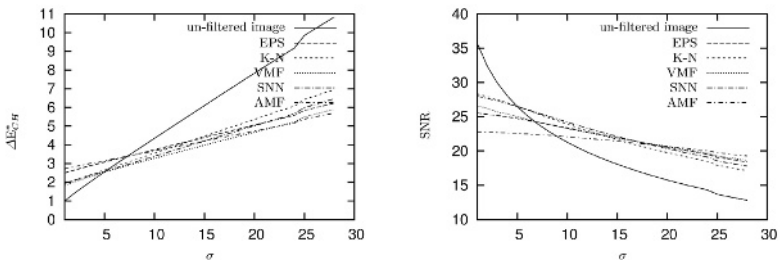

Fig. 6. Example image "`peppers`" corrupted by additive Gaussian noise

the measurements. For example, the arithmetic mean filter is judged differently for impulsive noise as it yields similar SNR but considerably different $\Delta \mathrm{E}_{CH}^{*}$.

As a conclusion we realize that better looking images (as quantitatively judged by $\Delta \mathrm{E}_{CH}^{*}$) will not always be best suited for further processing, as they may contain less information (as quantitatively judged by SNR) than images appearing worse visually.

References

1. Jaakko Astola, Pekka Haavisto, and Yrjo Neuvo. Vector median filters. *Proceedings of the IEEE*, 78:678–689, 1990.
2. Peter Bakker, L.J. van Fliet, and Piet W. Verbeek. Edge preserving orientation adaptive filtering. In *Proc. 5th Annual Conference of the Advanced School for Computing and Imaging*, pages 207–213, 1999.
3. Serge Chastel, Guido Schwab, and Dietrich Paulus. Web interface for image processing algorithms. In Simone Santini and Raimundo Schettini, editors, *Internet Imaging V* San Jose, 1 2004. Proc. of SPIE. volume 5304.
4. Rolf Gierling. *Farbmanagement*. Moderne Industrie Buch AG & Co. KG, Bonn 2001.
5. Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Prentice Hall, second edition, 2001.
6. David Harwood, Murali Subbarao, H. Hakalahti, and L. Davis. A new class of edge-preserving smoothing filters. *Pattern Recognition Letters*, 5:155–162, 1987.
7. M. Kuwahara, K. Hachimura, S. Eiho, and M. Kinoshita. *Digital Processing of Biomedical Images*, chapter Processing of ri-angiocardiographic images, pages 187–202. Plenum Press, New York, USA, 1976.
8. M. Nagao and T. Matsuyama. Edge preserving smoothing. *Computer Graphics and Image Processing*, 9:394–407, 1979.
9. M. Pietikainen and David Harwood. *Advances in Image Processing and Pattern Recognition*, chapter Segmentation of color images using edge-preserving, pages 94–99. North Holland, Amsterdam, Netherlands, 1986.
10. Konstantinos N. Plataniotis and Anastasios N. Venetsanopoulos. *Color Image Processing and Applications*. Springer Verlag, 2000.
11. Volker Rehrmann, editor. *Erster Workshop Farbbildverarbeitung*, Universität Koblenz–Landau, 1995.
12. Manfred Richter. *Einführung in die Farbmetrik*. Walter de Gruyter, Berlin, New York, 2 edition, 1981.
13. Stephen J. Sangwine and R. E. N. Horne. *The Colour Image Processing Handbook*. Chapman Hall, London 1998.
14. Rein van den Boomgaard. Decomposition of the Kuwahara-Nagao operator in terms of linear smoothing and morphological sharpening. In *Proc. of the 6th International Symposium on Mathematical Morphology*, pages 283–292, 2002.
15. Günter Wyszecki and W. S. Stiles. *Color Science: Concepts and Methods, Quantitative Data and Formulas*. John Wiley & Sons, second edition, 1982.
16. Xuemei Zhang and Brian A. Wandell. Color image fidelity metrics evaluated using image distortion maps. *Signal Processing*, 70(3):201–214, 11 1998.