Mathematical Problems in Engineering
Volume 4, pp. 529–538
Reprints available directly from the publisher
Photocopying permitted by license only

Adaptive Filters for Color Image Processing

V. PAPANIKOLAOU [a], K.N. PLATANIOTIS [b,*] and
A.N. VENETSANOPOULOS [c]

[a] *University of Toronto, Toronto, M5S 3G4, ON, Canada;* [b] *School of Computer Science, Ryerson Polytechnic University, M5B 2K3, ON, Canada;* [c] *Digital Image Processing Laboratory, University of Toronto, M5S 3G4, ON, Canada*

(Received 23 February 1998)

The color filters that are used to attenuate noise are usually optimized to perform extremely well when dealing with certain noise distributions. Unfortunately it is often the case that the noise corrupting the image is not known. It is thus beneficial to know *a priori* the type of noise corrupting the image in order to select the optimal filter. A method of extracting and characterizing the noise within a digital color image using the generalized Gaussian probability density function (pdf) (B.D. Jeffs and W.H. Pun, *IEEE Transactions on Image Processing*, **4**(10), 1451–1456, 1995 and *Proceedings of the Int. Conference on Image Processing*, 465–468, 1996), is presented. In this paper simulation results are included to demonstrate the effectiveness of the proposed methodology.

Keywords: Generalized Gaussian; Color; Adaptive; Noise; Image

1 INTRODUCTION

There are many color image filters that are used for noise reduction. The filters utilize the interchannel signal and noise correlation to improve performance but they are often optimized for a specific type of noise. However in practice the noise statistics within an image vary

* Corresponding author. Tel.: 416 979 5000, ext. 7062. Fax: 416 979 5064.
E-mail: kplatani@acs.ryerson.ca.

from application to application and they even vary in the same application from one image to the next [3,4]. Knowing the type of noise corrupting the image *a priori* would allow an appropriate filter to be chosen. This paper focuses on determining the underlying noise distribution corrupting an image utilizing the generalized Gaussian pdf [1,2]. Through knowledge of the noise distribution an optimal filter can then be chosen.

2 THE GENERALIZED GAUSSIAN PDF

The generalized Gaussian pdf is defined as [1]:

$$f_X(x,\gamma,\mu,\beta) = \frac{\gamma}{2\beta\Gamma(1/\gamma)}\exp\left\{-\left(\frac{|x-\mu|}{\beta}\right)^{\gamma}\right\} \quad \gamma > 0, \ \ x \in \Re \qquad (1)$$

The parameters β, and μ control the standard deviation and mean respectively while the value of γ, determines the decay rate of the density function and is thus an indication of the shape. The function $\Gamma(\cdot)$ is the well known gamma function [5]. The flexibility of generalized Gaussian pdf is illustrated by the fact that it can model a variety of pdfs.

From Table I it is evident that the generalized Gaussian pdf can model a wide range of noise distributions. It follows that given the noise within an image, the generalized Gaussian pdf can be used to determine an estimate of the noise distribution and the value of γ can be used to determine the type of distribution. Knowledge of the noise pdf provides the system the ability to select a filter that is known to be optimal in suppressing that type of noise in an image.

TABLE I Distributions of generalized Gaussian function

γ	β	μ	$f_x(x)$
2		0	Normal distribution
1	1	0	Laplacian distribution
$(0,1)$			Heavy tailed distributions
$(1,2)$			Mixture of normal and laplacian distribution
∞			Uniform distribution

3 THE NEW FILTER

The system shown in Fig. 1 illustrates how the image was filtered using the generalized Gaussian pdf.

As can be seen, the system is essentially a decision directed digital image filter. The last block has two inputs: γ and the corrupted image. The adaptive filter block takes γ, decides which filter is optimal against that type of noise and then applies that selected filter to the entire corrupted image. Table II illustrates what filter could be chosen according to the value of γ.

In the simulations the system that was used to extract the noise from the image is shown in Fig. 2.

Let $\mathbf{s}_{ij}$ be a 3 dimensional vector representing the original image, where i, j represent the spatial co-ordinates of the pixel. Let W represent a randomly placed window with dimensions $N \times N$. Those pixels that are within the boundaries specified by W are represented by $\mathbf{x}_{ij}$. Let $\mathbf{y}_{ij}$ represent the output of a digital filter to the input $\mathbf{x}_{ij}$. The noise extracted from the image is represented by $\mathbf{n}_{ij}$ and is the difference between $\mathbf{x}_{ij}$ and $\mathbf{y}_{ij}$.

The key component of this system is the filter. The filter's output, $\mathbf{y}_{ij}$ is an approximation to the original, uncorrupted image. The noise is represented by the difference between the approximate original image

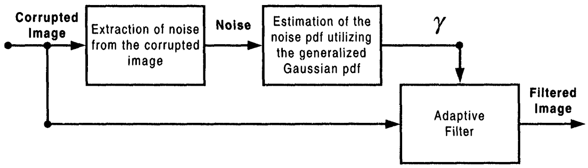

FIGURE 1 System used to filter a corrupted image using the generalized Gaussian pdf.

TABLE II Filters chosen according to γ

γ	*Noise pdf*	*Selected filter*
$[2, \infty)$	Gaussian–Uniform	AMF
$(1, 2)$	Mixed Gaussian and impulsive	FVF [4] using euclidean distance as the distance metric
$(0, 1]$	Long tailed	VMF using euclidean distance as the distance metric

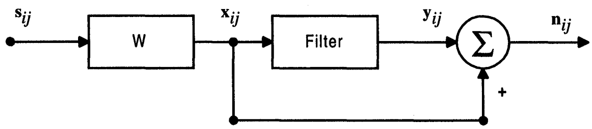

FIGURE 2 System utilizing a filter to extract the noise from the image.

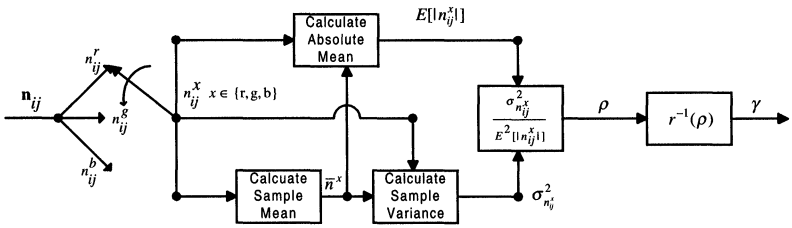

FIGURE 3 System used to determine γ.

$\mathbf{y}_{ij}$ and the corrupted image $\mathbf{x}_{ij}$ assuming additive noise. At this stage of the process, the value of γ is not known thus a filter must be chosen beforehand that performs fairly well against all types of noise. One such filter is the Arithmetic Mean Filter (AMF) [4]. This was used almost exclusively in the simulations but the Vector Media Filter (VMF) [4] was also used for comparison.

With the noise extracted it is now possible to obtain an estimate of the noise pdf. The system used is shown in Fig. 3.

The equations describing the variables introduced in Fig. 3 are:

$$E[|n^x_{ij}|] = \frac{1}{N^2} \sum_{(i,j)\in W} |n^x_{ij} - \bar{n}^x| \tag{2}$$

$$\bar{n}^x = \frac{1}{N^2} \sum_{(i,j)\in W} n^x_{ij} \tag{3}$$

$$\sigma^2 n^x_{ij} = \frac{1}{N^2 - 1} \sum_{(i,j)\in W} (n^x_{ij} - \bar{n}^x)^2 \tag{4}$$

$$\rho = \frac{\sigma^2 n^x_{ij}}{E^2[|n^x_{ij}|]} \tag{5}$$

$$r(\gamma) = \rho = \frac{\Gamma(1/\gamma)\Gamma(3/\gamma)}{\Gamma^2(2/\gamma)} \tag{6}$$

The $E[\cdot]$, σ, and ρ represent the expectation opererator, standard deviation and correlation coefficient [5]. It was assumed that the use of only one of the primary colors would provide an accurate enough estimate of the noise process corrupting the image. The system is flexible in the sense that any one of the three primary colors can be used to find γ. A lookup table was used to find $r^{-1}(\rho)$ [6] due to its complicated nature.

4 SIMULATION RESULTS

To assess the effectiveness of the systems they were tested on three types of noise distributions. These noise distributions are shown in Table III.

The proposed filter was tested by changing:

(i) Size of window
(ii) The noise corrupting the image (see Table III)
(iii) The image ("lenna" and "peppers" was used)
(iv) The filter used in the extraction of the noise
(v) The location of the window within the image
(vi) The color used to calculate γ (see Fig. 3)

The results are tabulated in Tables IV–IX. The values of γ were calculated by using the following procedure.

(i) Windows were randomly placed on the corrupted image being analyzed
(ii) At each of these window locations γ was calculated
(iii) The sample mean and standard deviation of these values was calculated

TABLE III Noise distributions used to test the systems

Noise	*Description of noise distribution*	*Optimal filter*	γ
Noise I	Gaussian noise with $\sigma = 30$ (standard deviation) and $\rho = 0.5$	AMF	2
Noise II	4% Impulsive noise with $\rho = 0.5$	VMF	1
Noise III	Mixed Gaussian ($\sigma = 30$ and $\rho = 0.5$) and 4% impulsive noise	FVF	(1, 2)

TABLE IV Lenna image corrupted with Noise I $\gamma_{\text{theoretical}} \in [2, \infty)$ for AMF

Color	32 × 32 *pixel window size*		64 × 64 *pixel window size*		128 × 128 *pixel window size*	
	$\gamma = \langle\gamma\rangle \pm \text{std}(\gamma)$	% *Conf.*	$\gamma = \langle\gamma\rangle \pm \text{std}(\gamma)$	% *Conf.*	$\gamma = \langle\gamma\rangle \pm \text{std}(\gamma)$	% *Conf.*
Red	2.203 ± 0.3073	76	2.150 ± 0.1315	89	2.126 ± 0.0474	100
Green	2.058 ± 0.2123	60	2.029 ± 0.0859	55	2.049 ± 0.0360	94
Blue	1.996 ± 0.1231	53	1.993 ± 0.617	41	1.995 ± 0.0270	39

TABLE V Lenna image corrupted with Noise II $\gamma_{\text{theoretical}} \in (0, 1]$ for VMF

Color	32 × 32 *pixel window size*		64 × 64 *pixel window size*		128 × 128 *pixel window size*	
	$\gamma = \langle\gamma\rangle \pm \text{std}(\gamma)$	% *Conf.*	$\gamma = \langle\gamma\rangle \pm \text{std}(\gamma)$	% *Conf.*	$\gamma = \langle\gamma\rangle \pm \text{std}(\gamma)$	% *Conf.*
Red	0.417 ± 0.0556	100	0.420 ± 0.0418	100	0.416 ± 0.0231	100
Green	0.468 ± 0.045	100	0.474 ± 0.0305	100	0.474 ± 0.022	100
Blue	0.500 ± 0.407	100	0.497 ± 0.0355	100	0.506 ± 0.0304	100

TABLE VI Lenna image corrupted with Noise III $\gamma_{\text{theoretical}} \in (1, 2)$ for FVF

Color	32 × 32 *pixel window size*		64 × 64 *pixel window size*		128 × 128 *pixel window size*	
	$\gamma = \langle\gamma\rangle \pm \text{std}(\gamma)$	% *Conf.*	$\gamma = \langle\gamma\rangle \pm \text{std}(\gamma)$	% *Conf.*	$\gamma = \langle\gamma\rangle \pm \text{std}(\gamma)$	% *Conf.*
Red	1.060 ± 0.1495	67	1.065 ± 0.1043	71	1.049 ± 0.0504	83
Green	1.157 ± 0.1186	91	1.145 ± 0.0880	93	1.322 ± 0.0422	100
Blue	1.224 ± 0.0793	100	1.227 ± 0.0462	100	1.221 ± 0.0262	100

TABLE VII Peppers image corrupted with Noise I $\gamma_{\text{theoretical}} \in [2, \infty)$ for AMF

Color	32 × 32 *pixel window size*		64 × 64 *pixel window size*		128 × 128 *pixel window size*	
	$\gamma = \langle\gamma\rangle \pm \text{std}(\gamma)$	% *Conf.*	$\gamma = \langle\gamma\rangle \pm \text{std}(\gamma)$	% *Conf.*	$\gamma = \langle\gamma\rangle \pm \text{std}(\gamma)$	% *Conf.*
Red	2.080 ± 0.1697	64	2.0614 ± 0.868	83	2.058 ± 0.0527	77
Green	2.037 ± 0.2612	69	2.012 ± 0.2029	76	2.043 ± 0.1354	74
Blue	2.088 ± 0.2874	64	2.009 ± 0.1568	44	2.027 ± 0.0536	83

The percentage confidence is defined as the percentage of the estimated γ values that selected the optimal filter to suppress the noise. For example, the percentage of γ values in the range $[2, \infty)$ is the % confidence for images corrupted with Gaussian noise.

TABLE VIII Peppers image corrupted with Noise II $\gamma_{\text{theoretical}} \in (0, 1]$ for VMF

Color	32 × 32 *pixel window size*		64 × 64 *pixel window size*		128 × 128 *pixel window size*	
	$\gamma = \langle\gamma\rangle \pm \text{std}(\gamma)$	% *Conf.*	$\gamma = \langle\gamma\rangle \pm \text{std}(\gamma)$	% *Conf.*	$\gamma = \langle\gamma\rangle \pm \text{std}(\gamma)$	% *Conf.*
Red	0.491 ± 0.0556	100	0.489 ± 0.0357	100	0.486 ± 0.0287	100
Green	0.455 ± 0.0643	100	0.450 ± 0.0375	100	0.452 ± 0.0236	100
Blue	0.457 ± 0.0664	100	0.460 ± 0.543	100	0.462 ± 0.0287	100

TABLE IX Peppers image corrupted with Noise III $\gamma_{\text{theoretical}} \in (1, 2)$ for FVF

Color	32 × 32 *pixel window size*		64 × 64 *pixel window size*		128 × 128 *pixel window size*	
	$\gamma = \langle\gamma\rangle \pm \text{std}(\gamma)$	% *Conf.*	$\gamma = \langle\gamma\rangle \pm \text{std}(\gamma)$	% *Conf.*	$\gamma = \langle\gamma\rangle \pm \text{std}(\gamma)$	% *Conf.*
Red	0.438 ± 0.0674	0	0.429 ± 0.0417	0	0.424 ± 0.0334	0
Green	0.459 ± 0.0659	0	0.450 ± 0.0393	0	0.452 ± 0.0269	0
Blue	0.463 ± 0.0645	0	0.461 ± 0.0521	0	0.461 ± 0.0314	0

From the experiments the following conclusions can be drawn:

- From Tables IV through IX it can be seen that by changing the size of the window changed the results slightly. From these numbers it can be seen that the mean value of γ decreases slightly with an increasing window. This small change may be attributed to the increased edge activity as the window size increased. The standard deviation decreased with an increasing window size and in general the percentage confidence increased with larger window sizes.
- From the simulations it was found that the AMF was superior to the VMF as the filter in Fig. 2 since the AMF selected the appropriate filter many more times the VMF.
- The filter worked extremely well with the images corrupted by Noise I and Noise II since it chose the optimal filter the majority of the time.
- For the lenna image corrupted with Noise III the Fuzzy Vector Filter (FVF) was selected the majority of the time (see Table VI) but the same is not true of the peppers image corrupted with the same type of noise (see Table IX). This indicates that the system was sensitive to the specific image being analyzed for the mixture of Gaussian and impulsive noise.

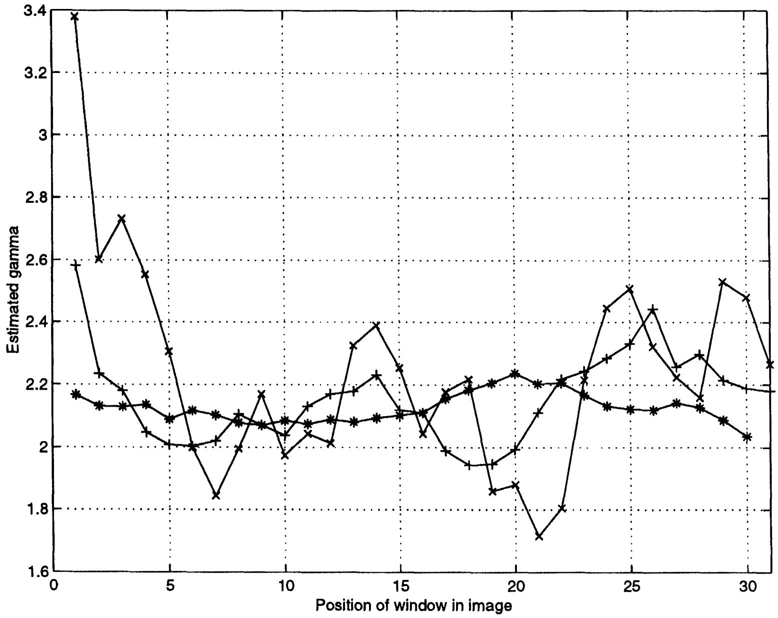

FIGURE 4 Variation of γ within image.

- Figure 4 illustrates the variation of γ within an image. This plot is indicative of all the experiments. From this plot it is evident that the value of γ is not invariant to the position of the window but the fluctuations in γ decreases as window size increases. Thus a larger window size can be used to minimize the fluctuations.
- The noise in the image was known to have the same distribution among the three color channels. This implies that the γ should reflect this and thus produce equivalent results in each of the color spaces. The results in Tables IV–IX illustrate that this was the case.
- Figure 5 contains the results of filtering the lenna image corruped with Noise II with the VMF and AMF. The filter correctly selected the VMF filter γ 100% of the time (see the Tables V and VIII). For comparison purposes the AMF has also been shown. This is used to represent the output a fictitious system that has no knowledge of the noise distribution. The NMSE was used to compare the two results due to its general acceptance and tractability.

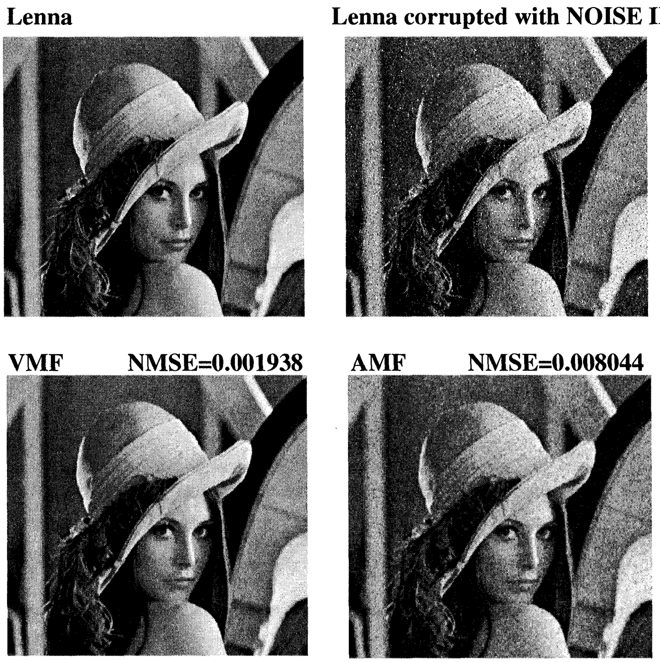

FIGURE 5 Results of filtering with VMF and AMF.

The NMSE is given by:

$$NMSE = \frac{\sum_{i=0}^{N1} \sum_{j=0}^{N2} \|y(i,j) - y'(i,j)\|}{\sum_{i=0}^{N1} \sum_{j=0}^{N2} \|y(i,j)\|} \tag{7}$$

where N_1, and N_2 are the image dimensions, (i, j) are the pixel coordinates and $y(i, j)$ and $y'(i, j)$ are the original corrupted image and the filtered image respectively. From the results the NMSE criterion suggests that the VMF is better and thus it can be concluded that the system selected the optimal filter.

5 CONCLUSIONS

A system that estimate the noise pdf and utilizes this to select the optimal filter has been presented in this paper. The performance of the system was assessed using various multichannel noise distributions. The system illustrated that the majority of the time it selected the optimal filter for Gaussian and impulsive noise. The system was slightly sensitive to the position of the window but simulations illustrated that a larger window size could circumvent this problem. Alternative methods of determining the shape parameter are currently under consideration.

References

[1] B.D. Jeffs and W.H. Pun, Adaptive image restoration using a generalized gaussian model for unknown noise, *IEEE Transactions on Image Processing*, **4**(10), 1451–1456, 1995.

[2] B.D. Jeffs and W.H. Pun, Simple shape parameter estimation from blurred observations for a generalized Gaussian MRF image prior used in MAP image restoration, *Proceedings of the Int. Conference on Image Processing*, ICP-96, pp. 465–468, 1996.

[3] I. Pitas, *Digital Image Processing Algorithms*, Prentice Hall, 1993.

[4] I. Pitas and A.N. Venetsanopoulos, *Nonlinear Digital Filters: Principles and Applications*, Kluwer Academic, Norwell Ma., 1990.

[5] A.L. Garcia, *Probability and Random Processes for Electrical Engineering*, Addison Wesley, USA, 1994.

[6] K. Sharifi and A.L. Garcia, Estimation of shape parameter for generalized gaussian distributions in subband decompositions of video, *IEEE Transactions on Circuits and Systems for Video Technology*, **5**(1), 52–55, 1995.

Mathematical Problems in Engineering

Special Issue on Modeling Experimental Nonlinear Dynamics and Chaotic Scenarios

Call for Papers

Thinking about nonlinearity in engineering areas, up to the 70s, was focused on intentionally built nonlinear parts in order to improve the operational characteristics of a device or system. Keying, saturation, hysteretic phenomena, and dead zones were added to existing devices increasing their behavior diversity and precision. In this context, an intrinsic nonlinearity was treated just as a linear approximation, around equilibrium points.

Inspired on the rediscovering of the richness of nonlinear and chaotic phenomena, engineers started using analytical tools from "Qualitative Theory of Differential Equations," allowing more precise analysis and synthesis, in order to produce new vital products and services. Bifurcation theory, dynamical systems and chaos started to be part of the mandatory set of tools for design engineers.

This proposed special edition of the *Mathematical Problems in Engineering* aims to provide a picture of the importance of the bifurcation theory, relating it with nonlinear and chaotic dynamics for natural and engineered systems. Ideas of how this dynamics can be captured through precisely tailored real and numerical experiments and understanding by the combination of specific tools that associate dynamical system theory and geometric tools in a very clever, sophisticated, and at the same time simple and unique analytical environment are the subject of this issue, allowing new methods to design high-precision devices and equipment.

Authors should follow the Mathematical Problems in Engineering manuscript format described at http://www.hindawi.com/journals/mpe/. Prospective authors should submit an electronic copy of their complete manuscript through the journal Manuscript Tracking System at http://mts.hindawi.com/ according to the following timetable:

Manuscript Due	December 1, 2008
First Round of Reviews	March 1, 2009
Publication Date	June 1, 2009

Guest Editors

José Roberto Castilho Piqueira, Telecommunication and Control Engineering Department, Polytechnic School, The University of São Paulo, 05508-970 São Paulo, Brazil; piqueira@lac.usp.br

Elbert E. Neher Macau, Laboratório Associado de Matemática Aplicada e Computação (LAC), Instituto Nacional de Pesquisas Espaciais (INPE), São Josè dos Campos, 12227-010 São Paulo, Brazil ; elbert@lac.inpe.br

Celso Grebogi, Center for Applied Dynamics Research, King's College, University of Aberdeen, Aberdeen AB24 3UE, UK; grebogi@abdn.ac.uk