

Fuzzy Anisotropic Smoothing of Scalar Images

Bekir DIZDAROGLU

Department of Computer Engineering, Karadeniz Technical University
Trabzon, 61080, Turkey

ABSTRACT

This study suggests a fuzzy anisotropic smoothing technique on scalar (grayscale) images using partial differential equations (PDEs) and fuzzy sets. This technique preserves the image details such as edges while noise is being reduced. The PDE and fuzzy rule based techniques are employed in approximation of smoothing coefficients calculated based on structure and noisiness information. Utilizing multiple tools constitutes the advantages of the proposed technique compared to the others. The results reveal that this technique can perfectly be used in denoising images in which fine details are definitely to be preserved during the smoothing operation.

Keywords: Anisotropic Smoothing, Type I and Type II Fuzzy Sets, Structure Measure, Noisiness Measure.

1. INTRODUCTION

Recent image and video processing researches widely employ PDEs and fuzzy sets. Some fundamental improvements in terms of inpainting, colorization, segmentation, registration, restoration and editing, have been achieved by employing PDEs and fuzzy logic techniques in image processing [1-13]. The experiments show that fuzzy sets and PDEs improve the performance and produce better results.

PDEs based denoising techniques can be considered as *non-linear filters* smoothing the image gradually by minimizing the image variations [1, 3]. In the literature there are quite number of techniques based on PDEs and fuzzy sets for denoising images [8-12]. Type I fuzzy sets were employed by Song *et al.* [8] and Aja *et al.* [9]. Besides, type II fuzzy sets were used by Puvanathan *et al.* [10] for image denoising. Schulte *et al.* [11-12] suggested detection and filtering methods based on type I fuzzy sets to reduce noise on images.

This paper suggests a different algorithm based on fuzzy anisotropic smoothing approach. Unlike other methods using gradient information directly, the introduced algorithm employs the smoothed image structure and the noise distribution measures. This provides a flexible and robust way to reduce the noise on the scalar images. Although the approach suggested in [10] looks similar to the proposed algorithm, more precise image structure and noise distribution are employed in this work, and thus the generated results are better compared to those of previous ones.

Isotropic smoothing

Isotropic smoothing is a fine way to denoise the images [1]. All the restoration methods, starting from Tikhonov in [4] to the classical linear filtering of images, result similarly in terms of regularization behavior. The smoothing process is shown below:

Let $I_{noisy}: \Omega \rightarrow \mathbb{R}$ be a noisy 2D scalar image to be denoised. There are three major noise types which are *impulse noise*, *additive noise* and *multiplicative noise*. Among these, additive noise model is an independent noise type that can be described by Gaussian distribution. Arising because of randomness superimposed on the image under process, *Gaussian distribution* is a very good example of the noise that occurs in many cases [11-12]. That is the reason why Gaussian noise is preferred for analyses in this study. This type of noise can be seen as high frequency variations n with low magnitude in pixels of the original image [1]:

$$I_{noisy} = I_{original} + n \quad (1)$$

Denoising operation of I_{noisy} is to minimize the variations in the image. Eq. (2) is used for minimization operation on the image [4]:

$$\inf_{I: \Omega \rightarrow \mathbb{R}} E_{Tikhonov}(I) = \int_{\Omega} \|\nabla I\|^2 d\Omega \quad (2)$$

As shown in Eq. (3), the *image gradient* denoted by ∇I is the derivation of *scalar image* I with respect to its spatial coordinates $\mathbf{p} = (x, y)$:

$$\nabla I = (I_x, I_y)^T = \left(\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y} \right)^T \quad (3)$$

In order to represent magnitudes of the scalar image I and its maximum variation directions, a vector $\nabla I: \Omega \rightarrow \mathbb{R}^2$ is created. Scalar and pointwise measure of the image variations are given by the *gradient norm* $\|\nabla I\|$ which is used in image analysis in many cases:

$$\|\nabla I\| = \sqrt{I_x^2 + I_y^2} \quad (4)$$

Finding the function I minimizing the functional $E(I)$ is not an easy task. Necessary condition is given by the *Euler-Lagrange equations*, which must be confirmed by I to reach a minimum of $E(I)$:

$$\frac{\partial E}{\partial I} = \frac{\partial F}{\partial I} - \frac{d}{dx} \frac{\partial F}{\partial I_x} - \frac{d}{dy} \frac{\partial F}{\partial I_y} = 0 \quad (5)$$

$$\text{where } F = \|\nabla I\|^2 = \left(\sqrt{I_x^2 + I_y^2} \right)^2.$$

A classic iterative method called *gradient descent* is employed to solve PDE in Eq. (5). As a matter of fact, Eq. (5) can be seen as the gradient of the functional $E(I)$. A local minimizer I_{min} of $E(I)$ can be found by starting from I_0 and then following the opposite direction of the gradient:

$$\begin{cases} I_{(t=0)} = I_{noisy} \\ \frac{\partial I}{\partial t} = - \left(\frac{\partial F}{\partial I} - \frac{d}{dx} \frac{\partial F}{\partial I_x} - \frac{d}{dy} \frac{\partial F}{\partial I_y} \right) \end{cases} \quad (6)$$

Eq. (7) is reached as the solution of Eq. (2):

$$\frac{\partial I}{\partial t} = I_{xx} + I_{yy} = \Delta I \quad (7)$$

where Δ is Laplace operator. This kind of PDE is called a *diffusion* or *heat equation*.

Basically, Eq. (7) at a particular time t gives the convolution of I_{noisy} with a *normalized 2D Gaussian kernel* G_σ of variance $\sigma = \sqrt{2t}$: $I_{noisy} * G_\sigma$ means the *linear* smoothing where $G_\sigma = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$.

As seen above clearly, during the PDE evaluation, the image is blurred gradually in an *isotropic way*. Here, isotropic smoothing acts as a *low-pass filter* suppressing high frequencies in the image I . Unfortunately, since image edges and noises are both high frequency signals, the edges are quickly blurred by this operation (See Figure 1). Especially in image restoration, non-linear and anisotropic smoothing methods should be found for a better result.



Figure 1: Result of Heat equation method applied on Cameraman's image which is artificially corrupted with additive Gaussian noise ($\sigma = 20$) after 100 iterations.

Anisotropic smoothing

The limitations of linear methods leading to isotropic smoothing were eliminated by Perona and Malik [5] proposed a nonlinear form of the Eq. (7). This idea is derived from divergence form of equation as shown in Eq. (8).

$$\frac{\partial I}{\partial t} = \Delta I = \text{div}(\nabla I) \quad (8)$$

The smoothing process can be controlled more precisely by adding a function $g(\|\nabla I\|)$ delimited by [0,1] in the divergence form:

$$\frac{\partial I}{\partial t} = \text{div}(g(\|\nabla I\|)\nabla I) \quad (9)$$

In order to stop the diffusion $g: \mathbb{R} \rightarrow \mathbb{R}$ is defined as a decreasing function vanishing on the edges (high gradients) for an anisotropic smoothing, and the function is close to 1 on homogenous areas (low gradients) for an isotropic smoothing. Perona-Malik proposed:

$$g(\|\nabla I\|) = \exp\left(-\frac{\|\nabla I\|^2}{K^2}\right) \quad (10)$$

where K is a threshold that helps differentiate homogeneous regions and edges.

A decomposed form of this equation has been proposed in [1, 6] to see how exactly the PDE in Eq. (9) behaves:

$$\frac{\partial I}{\partial t} = c_\xi I_{\xi\xi} + c_\eta I_{\eta\eta} \quad (11)$$

where $c_\xi = g(\|\nabla I\|)$ and $c_\eta = g'(\|\nabla I\|)\|\nabla I\| + g(\|\nabla I\|)$.

To give an example, using the proposed function in Eq. (10), the following results are found [1]:

$$c_\xi = \exp\left(-\frac{\|\nabla I\|^2}{K^2}\right) \text{ and } c_\eta = \exp\left(-\frac{\|\nabla I\|^2}{K^2}\right) \left(1 - 2\frac{\|\nabla I\|^2}{K^2}\right).$$

$I_{\xi\xi}$ and $I_{\eta\eta}$ are the second derivatives of I in orthogonal directions ξ and η , and can be seen as 1D oriented Laplacian:

$$I_{\xi\xi} = \frac{\partial^2 I}{\partial \xi^2} = \xi^T \mathbf{H} \xi \quad (12)$$

$$I_{\eta\eta} = \frac{\partial^2 I}{\partial \eta^2} = \eta^T \mathbf{H} \eta \quad (13)$$

where \mathbf{H} is the *Hessian matrix* of I :

$$\mathbf{H} = \begin{pmatrix} I_{xx} & I_{xy} \\ I_{yx} & I_{yy} \end{pmatrix} \quad (14)$$

Figure 2 depicts the unit vectors η and ξ which are defined by the *gradient direction* and its *orthogonal* respectively:

$$\eta = \frac{\nabla I}{\|\nabla I\|} = \frac{1}{\|\nabla I\|} (I_x, I_y)^T \quad (15)$$

$$\xi = \eta^\perp = \frac{1}{\|\nabla I\|} (I_y, -I_x)^T \quad (16)$$

The image is smoothed in the direction of the image contour ξ with a weight c_ξ , and in the direction of the gradient η with a weight c_η by Eq. (11) which indicates *two coexistent and oriented 1D heat flows*.

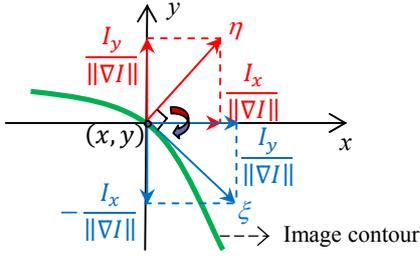


Figure 2: η and ξ vectors at point $\mathbf{p} = (x, y)$.

In this context $(\xi, \eta, c_\xi, c_\eta)$ shows the local diffusion geometry of the Perona-Malik approach. When $c_\xi \geq c_\eta$, anisotropic smoothing is conducted while preserving the image edges.

The test results show that the PDE operation of Eq. (9) better preserve the image properties than Eq. (7). Unfortunately, there still seems to be little noises in the smoothed image. However, a negative coefficient c_η is resulted by the function g proposed in Eq. (10), and inverse diffusion is sometimes produced as a consequence [1]. Inverse diffusion causes enhancements in image features. But it fails in an image with noise since the noise also becomes more evident as seen Figure 3. This is a drawback of the Perona-Malik method.



Figure 3: Result of Perona-Malik method applied on Cameraman's image which is artificially corrupted with additive Gaussian noise ($\sigma = 20$) after 100 iterations (The threshold value of the structure measure is chosen as 30).

2. THE PROPOSED METHOD

Fuzzy set theory is utilized in improving human understanding of the systems in terms of uncertainty and vagueness. The developed method based on the fuzzy sets theory enables us to minimize the effect of uncertainty in parameters used in anisotropic smoothing.

As shown in [10], fuzzy anisotropic smoothing algorithm is composed of five steps. Here, the steps are explained only for smoothing coefficient c_ξ as follows. The steps can be adopted to explain another coefficient c_η .

Step 1: Image properties corresponding to linguistic labels are set. Fuzzy operation is performed on each property by using the

proper membership function. The value of smoothing coefficient c_ξ is set as the largest in homogenous and noisy regions and smallest at edges. Therefore, if the image structure is low and the noise degree is high then the smoothing operation must be applied at point \mathbf{p} of the image and the coefficient should be set to a high value. Two linguistic variables can be defined according to image structure and noise degree, which are used in defining two fuzzy variables as the following:

Structure measure:

$$S = \|\nabla I_\sigma\| = \sqrt{\text{trace}(\mathbf{G}_\sigma)} \quad (17)$$

$$\mathbf{G}_\sigma = (\nabla I \nabla I^T) * G_\sigma \quad (18)$$

where $\text{trace}(\mathbf{G}_\sigma)$ stands for trace of the matrix \mathbf{G}_σ .

[11-12] defines the noisiness measure as a composition of two sub-parts both of which are used to define corrupted noisy pixels.

The first part aims at determining whether the image pixel $I(\mathbf{p})$ at point \mathbf{p} is a corrupted one or not. In order to understand this, a window of the dimension of $(2k + 1) \times (2k + 1)$ (k is set to 1) centered around $I(\mathbf{p})$ is first checked. Later, the mean differences n_1 in this window are calculated as follows:

$$n_1 = \frac{\sum_{i=-k}^k \sum_{j=-k}^k |I(x+i, y+j) - I(x, y)|}{(2k+1)^2 - 1} \quad (19)$$

High n_1 values are generally resulted by corrupted pixels, because noisy pixels are normally relatively more eye catching compared to the pixels in their neighborhood. High values of n_1 are also obtained in edge pixels. Therefore another value denoted as n_2 is also calculated:

$$n_2 = \frac{\sum_{i=-k}^k \sum_{j=-k}^k n_1(x+i, y+j)}{(2k+1)^2} \quad (20)$$

If both n_1 and n_2 are high values, the pixel is an edge pixel instead of noisy one. So when n_1 and n_2 values are close, it is thought that the pixel is noise free; otherwise, the pixel is a noisy one. The following fuzzy rule can be used in implementation of this approach:

Fuzzy Rule 1: Define if a central pixel $I(\mathbf{p})$ is corrupted with noise:

IF $N = |n_1 - n_2|$ is high

THEN the central pixel $I(\mathbf{p})$ is a noisy pixel.

Fuzzy Rule 1 is employed in calculation of the output of the first detection part by using the membership function μ^N .

Similar to the first detection part, the degree showing that a certain pixel $I(\mathbf{p})$ can be noisy is calculated for the second part. These parts complete each other to obtain a more robust detection method improving the global performance.

A window of dimension of $(2k + 1) \times (2k + 1)$ (k is set to 1) is chosen around $I(\mathbf{p})$. Eight neighbor pixels around $I(\mathbf{p})$ correspond to different directions {North West (NW), North (N), North East (NE), East (E), South East (SE), South (S), South West (SW), West (W)}. The following equation is defining the

gradient value $\nabla_{\mathbf{q}}I(\mathbf{p})$ of point \mathbf{p} in direction D , which is at point \mathbf{q} :

$$\nabla_{\mathbf{q}}I(\mathbf{p}) = I(\mathbf{p} + \mathbf{q}) - I(\mathbf{p}) \quad (21)$$

where \mathbf{q} and \mathbf{p} are used for one of the eight directions and for the center of the gradient respectively. Two cases of high gradient values occur among the eight basic calculated gradients. The first case is when one of the two pixels is noisy. The other case is when there is an edge. As a consequence of this, two related gradient values should be used to detect only the first case. These two related gradient values defined in the same direction as the basic gradients, are determined by the centers making a right-angle with the direction of the corresponding basic gradient. Figure 4 depicts this for the S-direction, i.e. for $\mathbf{q} = (1,0)$. Basic gradient and two related gradient values at point \mathbf{p} are stated as $\nabla_{\mathbf{q}}I(x, y)$, $\nabla_{\mathbf{q}}I(x, y - 1)$ and $\nabla_{\mathbf{q}}I(x, y + 1)$, respectively.

$\mathbf{q} = (i, j)$	-1	0	1	
-1	NW	N	NE	
0	W	$\mathbf{p} = (x, y)$	E	
1	SW	S	SE	
	Related	Basic	Related	

Figure 4: The centers used to calculate the related gradient values in S-direction.

An overview of the involved gradient values in each direction is given in Table 1.

Table 1: The gradient values to calculate the fuzzy gradient.

D	Basic ∇	Related ∇ s	
NW	$\nabla_{NW}I(\mathbf{p})$	$\nabla_{NW}I(x + 1, y - 1)$	$\nabla_{NW}I(x - 1, y + 1)$
N	$\nabla_NI(\mathbf{p})$	$\nabla_NI(x, y - 1)$	$\nabla_NI(x, y + 1)$
NE	$\nabla_{NE}I(\mathbf{p})$	$\nabla_{NE}I(x - 1, y - 1)$	$\nabla_{NE}I(x + 1, y + 1)$
E	$\nabla_EI(\mathbf{p})$	$\nabla_EI(x - 1, y)$	$\nabla_EI(x + 1, y)$
SE	$\nabla_{SE}I(\mathbf{p})$	$\nabla_{SE}I(x - 1, y + 1)$	$\nabla_{SE}I(x + 1, y - 1)$
S	$\nabla_SI(\mathbf{p})$	$\nabla_SI(x, y - 1)$	$\nabla_SI(x, y + 1)$
SW	$\nabla_{SW}I(\mathbf{p})$	$\nabla_{SW}I(x - 1, y - 1)$	$\nabla_{SW}I(x + 1, y + 1)$
W	$\nabla_WI(\mathbf{p})$	$\nabla_WI(x - 1, y)$	$\nabla_WI(x + 1, y)$

Finally, degrees τ_{noise}^D and τ_{free}^D in the type I fuzzy set *noise* and in the type I fuzzy set *noise-free* for each direction D are calculated. The Fuzzy Rules 2 and 3 are employed in calculating the degrees.

Fuzzy Rule 2: Define if a central pixel $I(\mathbf{p})$ is corrupted with noise for a certain direction D :

IF ($|\nabla_D I(\mathbf{p})|$ is **not high**) AND ($|\nabla'_D I(\mathbf{p})|$ is **high**) AND ($|\nabla''_D I(\mathbf{p})|$ is **high**)

OR

IF ($|\nabla_D I(\mathbf{p})|$ is **high**) AND [($|\nabla'_D I(\mathbf{p})|$ is **not high**) OR ($|\nabla''_D I(\mathbf{p})|$ is **not high**)]

THEN the central pixel $I(\mathbf{p})$ is a noisy pixel in direction D .

Fuzzy Rule 3: Define if a central pixel $I(\mathbf{p})$ is not corrupted with noise for a certain direction D :

IF ($|\nabla_D I(\mathbf{p})|$ is **high**) AND ($|\nabla'_D I(\mathbf{p})|$ is **high**) AND ($|\nabla''_D I(\mathbf{p})|$ is **high**)

OR

IF ($|\nabla_D I(\mathbf{p})|$ is **not high**) AND ($|\nabla'_D I(\mathbf{p})|$ is **not high**) AND ($|\nabla''_D I(\mathbf{p})|$ is **not high**)

THEN the central pixel $I(\mathbf{p})$ is not a noisy pixel for a certain direction D .

In the above rules, $\nabla_D I(\mathbf{p})$, and $\nabla'_D I(\mathbf{p})$ and $\nabla''_D I(\mathbf{p})$ denote basic gradient and two related gradient values, respectively. Conjunctions and disjunctions are contained in the rules. Triangular norms and co-norms are used to represent AND and OR operators in fuzzy logic. Here, the product and probabilistic sum are employed. Besides, the standard not operator $\text{not}(x) = 1 - x$, with $x \in [0,1]$ is also used as negation. “($|\nabla_D I|$ is **not high**) AND ($|\nabla'_D I|$ is **high**) AND ($|\nabla''_D I|$ is **high**)” is calculated by using the product triangular norm as: $(1 - \mu^{\text{high}}(|\nabla_D I|)) \cdot (\mu^{\text{high}}(|\nabla'_D I|)) \cdot (\mu^{\text{high}}(|\nabla''_D I|))$.

Figure 5 illustrates the type I fuzzy membership function μ^{high} . Optimal values for a and b parameters were found experimentally [11-12]. These parameters, formulated below, are optimal to distinguish noise and edges (contours).

$$a = \frac{\sum_{i=-k}^k \sum_{j=-k}^k |n_1(x + i, y + j) - n_1(x, y)|}{(2k + 1)^2 - 1} \quad (22)$$

$$b = 1.2a$$

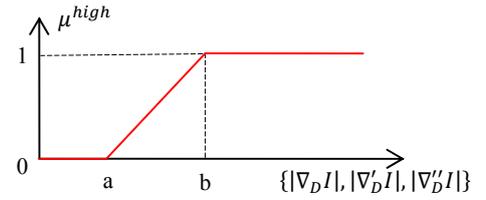


Figure 5: Type I fuzzy membership function μ^{high} .

Eight membership degrees in the type I fuzzy set for *noise* and *noise-free* situations shape the output of the second detection method for the eight neighboring pixels of the current point \mathbf{p} . Fuzzy Rules 2 and 3 are used to calculate the degrees $\tau_{noise}^D(I(\mathbf{p}))$ and $\tau_{free}^D(I(\mathbf{p}))$, respectively.

As shown in the following, both detection methods are combined to reset the noisiness measure N more precisely:

$$N = \begin{cases} 0, & \text{if } \sum_{D \in \{N, \dots, S\}} \tau_{noise}^D < \sum_{D \in \{N, \dots, S\}} \tau_{free}^D \\ \text{not changed,} & \text{otherwise} \end{cases} \quad (23)$$

Structure measure S is based on the smoothed gradient version of the image $I(\mathbf{p})$. [9-10] employed the standard deviation, which is calculated as the absolute of difference between the intensity of pixel and the average of its neighbor pixels, as a noisiness measure. However, this measure perceives some edges wrongly as noise in some cases as shown in Figure 6.b. Here, it should be stated that high noisiness measure at point \mathbf{p} shows a high noise degree. A normalization operation is applied on both structure and noisiness measures to set them between 0 and 1. These fuzzy variables make use of two corresponding

type II fuzzy sets. These type II fuzzy sets, called S and N , are related to the low structure measure and high noisiness measure respectively. Type II membership functions are depicted in Figure 7. Here, Eq. (10) is employed to obtain the form of the membership functions. The upper and lower membership functions are determined based on the study of Tizhoosh [13]. As opposed to the gradient norm at noise, it should be bigger than the threshold K at edges. Upper and lower membership functions can be defined as follows:

$$\mu_U = g^{1/\alpha} \quad (24)$$

$$\mu_L = g^\alpha \quad (25)$$

where $\alpha \in (1, \infty)$. In experiments, $\alpha \in (1, 2]$ has been used because setting $\alpha \gg 2$ is usually not significant for image processing [13].

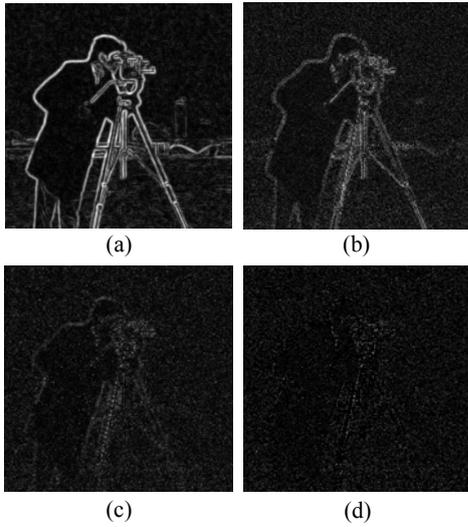


Figure 6: Measures were obtained from Cameraman's image corrupted with additive Gaussian noise ($\sigma = 20$): Proposed structure measure (a), noisiness measure presented in [9-10] (b), proposed noisiness measures of first (c) and second detection parts (d) for $k = 1$.

Step 2: A knowledge-base composed of Fuzzy Rule 4 is created in the second step of smoothing algorithm. Consequently the following fuzzy rule is obtained:

Fuzzy Rule 4: Define if c_ξ is a high value:

IF S is low AND N is high THEN c_ξ is a high value.

This indicates that higher smoothing coefficient at point \mathbf{p} is required when structure value is low and noise level is high.

Step 3: In order to meet the above rule the fuzzified inputs are combined. The following truth value is resulted by applying the product triangular norm to the fuzzy rule " S is a low value AND N is a high value":

$$\gamma_U = \mu_U^S(S) \cdot \mu_U^N(N) \quad (26)$$

$$\gamma_L = \mu_L^S(S) \cdot \mu_L^N(N) \quad (27)$$

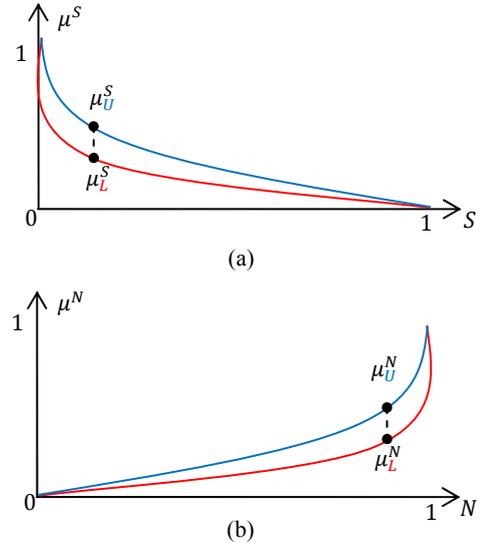


Figure 7: Type II fuzzy membership functions for the fuzzy variables that are "low structure" (a) and "high noisiness" (b) measures.

Two consequences are obtained for the above rule, each of which corresponds to the upper and lower limit of membership function. The smoothing degree of pixel at point \mathbf{p} is represented by these consequences. As a matter of fact activation degree of Fuzzy Rule 4 is represented by Eq. (26) and Eq. (27). This range of values (γ_U, γ_L) states the degree of membership in the fuzzy set "smoothing coefficient c_ξ is a high value". If the range of values is close to 1, the smoothing coefficient c_ξ is also close to 1, which indicates the maximum smoothing. If the value is close to 0 the smoothing coefficient c_ξ is close to 0, which indicates no smoothing operation is to be done.

Step 4: After obtaining the consequences from rules, they are combined to produce an output distribution range. This study does not apply the step 4 because it employs only the Fuzzy Rule 4.

Step 5: The smoothing coefficient c_ξ is obtained by type reducing and defuzzifying operation on output. In order to perform these operations, the average of the upper and the lower fuzziness values are used. Consequently, the smoothing coefficient c_ξ in type II fuzzy anisotropic smoothing is defined as:

$$c_\xi = \frac{\gamma_U + \gamma_L}{2} \quad (28)$$

Therefore, the fuzzy based coefficients are used in the fuzzy anisotropic smoothing as follows:

$$\frac{\partial I}{\partial t} = \begin{cases} c_{\xi} I_{\xi\xi} + I_{\eta\eta}, & \text{if measure} > \frac{K}{\sqrt{2}} \\ c_{\xi} I_{\xi\xi} + c_{\eta} I_{\eta\eta}, & \text{otherwise} \end{cases} \quad (29)$$

where measure stands for the structure and noisiness measures, and K is the threshold for them.

The denoising PDE shown in Eq. (28) is compatible with all properties expressed above:

$$\begin{cases} I_{(t=0)} = I_{noisy} \\ I_{(t+1)} = I_{(t)} + dt \frac{\partial I_{(t)}}{\partial t} \end{cases} \quad (30)$$

where dt stands for time step.

3. EXPERIMENTAL RESULTS

As seen in Figure 8.a-b, a synthetic Gaussian noise ($\sigma = 20$) is added to the scalar Cameraman's image with the dimension of 256x256 for tests. The methods are also applied on an originally degraded image with the dimension of 342x259 as seen in Figure 9.a. The proposed method is compared with isotropic smoothing and anisotropic smoothing methods, which are the most popular PDE denoising methods used in smoothing images. The comparison of the proposed method is conducted with these basic methods since others are somewhat derivatives of these methods [1]. The threshold value of the structure measure is chosen as 200 for the anisotropic smoothing method, and threshold values of the structure and noisiness measures are chosen as 0.75 and 1.5 for the proposed method, respectively. dt is set to 3 for all methods used in image smoothing, and α is chosen as 2 for the proposed method.

The denoised images generated by the proposed method and the previous ones are shown in Figure 8.c-e for 65 iterations. Additionally, the mean square error (MSE) and peak signal to noise ratio (PSNR) between the original and denoised images are displayed in Table 2. As can be seen in the table and denoised images, the best result is obtained by the proposed method.

The smoothing results of original degraded image are shown in Figure 9.b-d for 50 iterations, which prove that the proposed method is presenting a better performance compared to the others.

The methods were implemented in Microsoft Visual C++ 2005 by employing *CImg Library* [14]. The program was run on a PC with Pentium 2.20 GHz processor and 2 GB RAM.

4. CONCLUSION

In this paper, a method based on fuzzy anisotropic smoothing is presented for denoising scalar images. This method reduces the noise while preserving the image properties such as fine details. As proved with experimental results, the performance and output of the proposed method are promising. The results present good visual quality and numerical measures. This method can be applied to color images as a future task.

5. REFERENCES

- [1] D. Tschumperlé, "PDEs Based Regularization of Multivalued Images and Applications", **PhD Thesis**, The University of Nice Sophia Antipolis, 2002.
- [2] B. Dizdaroğlu and A. Gangal, "A Spatiotemporal Algorithm for Detection and Restoration of Defects in Old Color Films", **Lecturer Notes in Computer Sciences** 4678, 2007, pp. 509-520.
- [3] D. Tschumperlé and R. Deriche, "Vector-Valued Image Regularization with PDEs, A Common Framework for Different Applications", **IEEE Transactions on Pattern Analysis and Machine Intelligence** 27, No. 4, April 2005.
- [4] A. N. Tikhonov, "Regularization of Incorrectly Posed Problems", **Soviet. Math. Dokl.**, 4, 1963, pp. 1624-1627.
- [5] P. Perona and J. Malik, "Scale-space and Edge Detection Using Anisotropic Diffusion", **IEEE Transactions on Pattern Analysis and Machine Intelligence** 12, 7 (1990), pp. 629-639.
- [6] P. Kornprobst, R. Deriche and G. Aubert, "Nonlinear Operators in Image Restoration", **In Proceedings of the International Conference on Computer Vision and Pattern Recognition**, Puerto Rico, June 1997, pp. 325-331.
- [7] L. A. Zadeh, "Fuzzy Sets", **Information and Control** 8, 1965, pp. 338-353.
- [8] S. Aja, C. Alberola and J. Ruiz, "Fuzzy Anisotropic Diffusion for Speckle Filtering", **Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing**, 2001, pp. 1261-1264.
- [9] J. Song and H.R. Tizhoosh, "Fuzzy Anisotropic Diffusion: A Rule-based Approach", **7th World Multiconference on Systemics, Cybernetics and Informatics Proceedings**, Orlando, Florida, 2003, pp. 241-246.
- [10] P. Puvanathan and K. Bizheva, "Interval Type-II Fuzzy Anisotropic Diffusion Algorithm for Speckle Noise Reduction in Optical Coherence Tomography Images", **Optics Express** 17, 2 (2009), pp. 733-746.
- [11] S. Schulte, V. De Witte, M. Nachtegaal, D. Van der Weken, E.E. Kerre, "Fuzzy Random Impulse Noise Reduction Method", **Fuzzy Sets and Systems** 158, 3 (2007), 270-283.
- [12] S. Schulte, V. De Witte, M. Nachtegaal, D. Van der Weken, E.E. Kerre, "Histogram-Based Fuzzy Colour Filter for Image Restoration", **Image and Vision Computing** 25, 9 (2007), 1377-1390.
- [13] H. R. Tizhoosh, "Image Thresholding Using Type II Fuzzy Sets", **Pattern Recognition** 38, 2005, pp. 2363-2372.
- [14] D. Tschumperlé, The CImg Library: <http://cimg.sourceforge.net>, The C++ Template Image Processing Library.

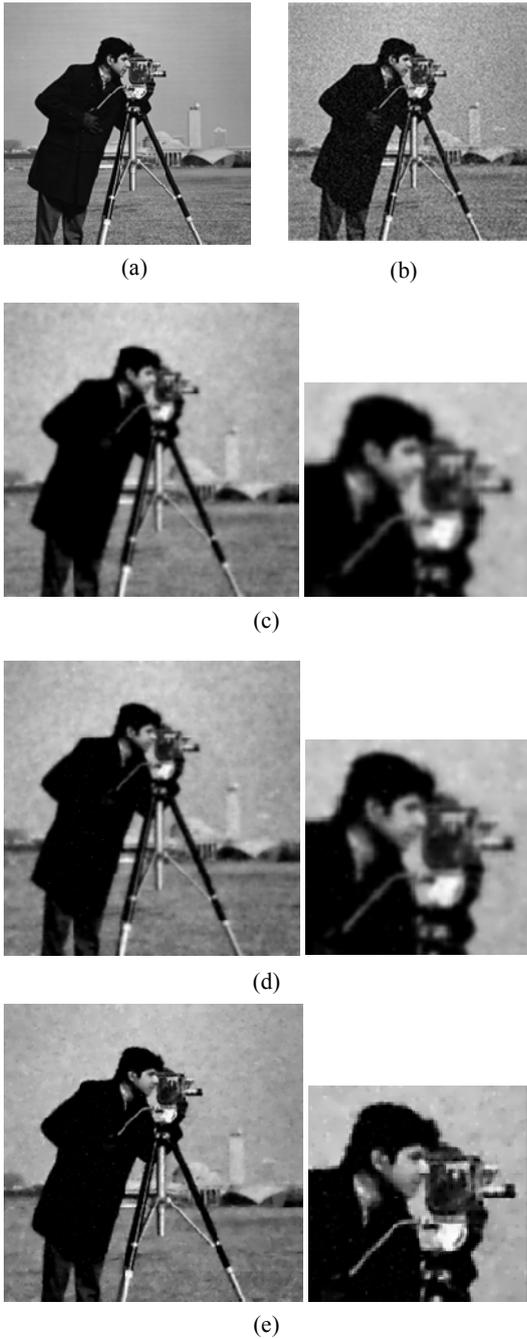


Figure 8: Original Cameraman's image (a), image corrupted with additive Gaussian noise (b), and isotropic smoothing method (c), anisotropic smoothing method (d) and proposed method (e).

Table 2: MSE and PSNR results.

Method	MSE	PSNR (dB)
Isotropic smoothing	415.032	21.949
Anisotropic smoothing	333.469	22.900
Proposed	225.939	24.590

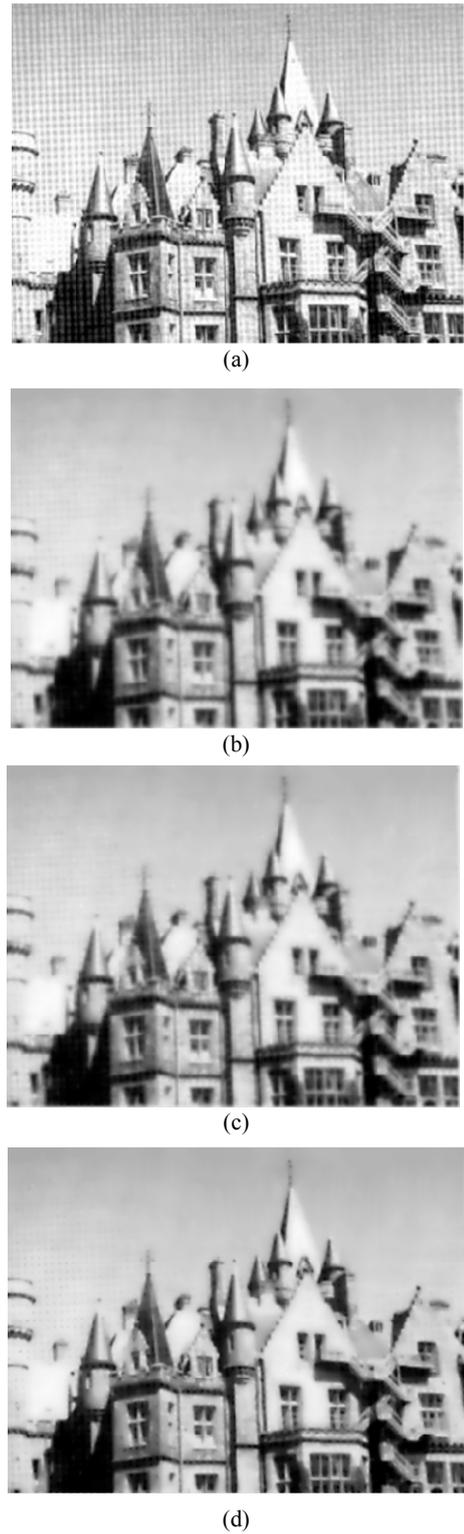


Figure 9: Original degraded image (a), isotropic smoothing method (b), anisotropic smoothing method (c) and proposed method (d).