PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA

MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

Faculty of New Information Technologies And Communication

ACADEMIC MASTER MEMORY

Field: computer science.

Specialty: Fundamental computer science.

Presented by:

- Gaba Samah.
- Nouali Roumaissa.

THEME

Image Restoration by Filtering (Perona-Malik Model)

Members of jury:

Dr Naima Merzougui	University of Ouargla	Supervisor
Dr Korichi Wassila	University of Ouargla	President
Dr Korichi Meriem	University of Ouargla	Examiner

College year: 2019/2020

ACKNOWLEDGEMENTS

First and foremost, we thank God for giving us the will and power to achieve this work.

We would like to testify, our deep gratitude and our sincere thanks to our supervisor **Dr N. Merzougui**, whose benevolent availability and her elaborate advice, made this memory possible.

Our thanks go to the members of the jury, who does not surely spare any effort for the effort and appreciation of this work.

We would also like to take this opportunity to thank to faculty for the enormous work carried out, the richness and quality of the teaching, for creating the most favorable conditions for the conduct of our studies.

May all those who have supported us with their meaningful advice, wishes and prayers finding here the explanation of our sincere gratitude's.

DEDICATION

I dedicate this modest work:

To my grandmother's pure soul ***Zohra*** my best friend and second mom “may God have mercy on her"

To my mother ***Bendahmane Naima,*** my precious gem, for her confidence and belief in me and in my abilities. Mum you’re my source of motivation, power, and happiness

To my father ***Mohammed Fouad*** for his sacrifices during my academic career, his patient, and his blessed prayers to me

To my one and only sister ***Affaf*** for her love and support, and my only brother ***Abderrahmane*** for always being there for me

I would like to thank my sister in law ***Hafida*** for her support and advices, it was very nice of you to guide me

I dedicate also this work and give special thanks to my grand-father ***Makhlouf*** (my second father) for his continuous prayers to me that don’t leave my ears, for his precious advices from his vast experience. May God prolong his life and preserve him for us

A special feeling of gratitude to my cousins ***Chahinez*** and ***Fatima Zohra***, my uncle ***Hamoud***, his wife ***Nora,*** and my aunt ***Ilhem*** for supporting me throughout the process. I will always appreciate all they have done

I dedicate this work and give special thanks to my partner in this thesis and life, my best friend ***Nouali Roumaissa;*** to her adorable family specially **Kheira** and **Tasnim** may God protect them all

To my lovely friends ***Majda, Youssra, Ghofrane, Randa*** for always being there for me when I need any support, you are my joy, May God keep you for me.

Special thanks to my supervisor Dr ***Merzougui.N*** for her great efforts

“Gaba Samah”

DEDICATION

A special feeling of appreciation to my loving parents ***Arous Fatiha*** my moonlight and ***Chouaib*** my good example in life

I dedicate this work to my three sisters ***Nahla***, ***Kheira***, and ***Afaf*** for the strength of their support they give me, specially my sisters in law ***Triki Manel*** the source of tenderness, and my lovely ***Ben Aissa Manel***

To my two brothers, ***Abdelkader Imed*** and ***Mohammed Riad*** for being always there for me

Also to my bestfriend and my soul twin ***Benjamaa Manar***

To my partner in this thesis my bestfriend and my world ***Gaba Samah***, to my second family ***Bendahmane Naima***, ***Gaba Mohammed Fouad***, my little sister ***Affaf*** and my brother ***Abderrahmane***

To my supervisor Dr ***Merzougui Naima***, thank you very much for your great efforts.

"Nouali Roumaissa"

Abstract

Image restoration refers to a group of methods or techniques that aim to eliminate or reduce the degradations that occurred in obtaining the digital image. In order to solve the problem of image degradation, we choose the Perona-Malik model, which is effective in eliminating noise while preserving edges and important information.

To better represent the results, we implemented the restoration's Perona-Malik process on a graphical interface and find out the correct values for its parameters. Also integrate several filters for a better comparison. We used the PSNR, SSIM and FSIM performance metrics to measure the quality of the resulting image and make the comparison. The quantitative results on noisy images prove the effectiveness and robustness of the method discussed compared to advanced methods. Noise is removed while edges and structural details are preserved.

Keywords:

Image processing, Image restoration, Perona-Malik, Anisotropic diffusion, PDE.

Résumé

La restauration d'image fait référence à un groupe de méthodes ou de techniques qui visent à éliminer ou à réduire les dégradations qui se sont produites lors de l'obtention de l'image numérique. Afin de résoudre le problème de la dégradation de l'image, nous choisissons le modèle Perona-Malik, qui est efficace dans l'élimination du bruit tout en préservant les bords et les informations importantes.

Pour mieux représenter les résultats, nous avons implémenté le procédé Perona-Malik de la restauration sur une interface graphique et en découvrons les bon valeurs pour ces paramètres.

Intégrez également d'autres modèles de pointe pour une meilleure comparaison. Nous avons utilisé les métriques de performances PSNR, SSIM et FSIM pour mesurer la qualité de l'image obtenue et faire la comparaison. Les résultats quantitatifs sur des images bruyantes prouvent l'efficacité et la robustesse de la méthode discutée par rapport aux méthodes de pointe. Le bruit est supprimé tandis que les bords et les détails structurels sont préservés.

Mots clés:

Traitement d'image, Restauration d'image, Perona-Malik, Diffusion anisotrope, PDE.

الـمـلـخـص

تشير استعادة الصورة إلى مجموعة من الأساليب أو التقنيات التي تهدف إلى إزالة أو تقليل التدهور الذي حدث في الحصول على الصورة الرقمية. من أجل حل مشكلة تدهور الصورة ، اخترنا نموذج Perona-Malik ، وهو نموذج فعال في إزالة الضوضاء مع الحفاظ على الحواف والمعلومات المهمة.
لتمثيل النتائج بشكل أفضل ، قمنا بتنفيذ عملية استعادة Perona-Malik على واجهة رسومية واكتشاف القيم الصحيحة لهذه المعلومات. قمنا أيضًا بدمج بعض النماذج الرائدة الأخرى للحصول على مقارنة أفضل. استخدمنا مقاييس أداء PSNR و SSIMو FSIM لقياس جودة الصور الناتجة وإجراء المقارنة. اثبتت النتائج الكمية على الصور المشوشة فعالية ومتانة الطريقة التي تمت مناقشتها مقارنة بالطرق المتقدمة. تم إزالة الضوضاء مع الحفاظ على الحواف والتفاصيل الهيكلية..

الكلمات المفتاحية:

معالجة الصور ، استعادة الصورة ، بيرونا مالك ، انتشار متباين الخواص ، PDE.

List of abbreviations:

PDE: Partial Differential Equations.

PM: Perona-Malik.

PSNR: Peak Signal to Noise Ration.

MSE: Mean Squared Error.

SSIM: Structured Similarity Indexing Method

FSIM: Feature Similarity Indexing Method

ESF: Edge Stopping Function.

dB: Decibel.

PDF: Probability Density Function.

List of tables:

Table of figures:

Contents

General Introduction

The image is the main source of information in many applications. However, fluctuations around the average intensities of the image, inherent either to the acquisition device (cameras, scanners, ultrasound machines, quantification, etc.) or to the scene itself (dust, bad weather, extreme conditions of lighting, ...), constitute degradations which affect the quality of the image and cause losses of information, these degradations are generally designated under the term of image noise. Restoring an image from a degraded image is a crucial problem and therefore of considerable importance.

The object of the restoration is to reduce, if not eliminate, the distortions introduced by the system used to acquire the image, most image restoration techniques often attempt to restore the image either linearly or non-linearly by reducing certain measures of degradation[1]. The principle is to model the noise to find the ideal, non-noisy image from the degraded image. As examples, the restoration is used in geophysics, biomedical imaging, in the media, in the restoration of deteriorated old films, etc...

The greyscale image is generally used in image processing. It contains all the important information for interpretation, decision-making with a grain of memory space and minimum calculation time.

Our aim in this thesis is to present study and apply a number of methods to solve the problem of greyscale image restoration. By focusing on the model of «Perona-Malik», and determining the best values of these parameters for good results.

Our thesis is organized as follows:

In the first chapter, we introduced the fundamental notions related to the representation of the digital image, generality about the image processing domain, while defining image restoration. In the second chapter, we discuss state-of-the-art of several methods in image restoration, their types, and properties. In the third chapter, we discuss the anisotropic diffusion "Perona-Malik" the first anisotropic PDE model was proposed by Perona-Malik in year 1990[2], discuss its purpose for eliminating noise and preserving contour at the same time. The last chapter devoted to the comparison of the results obtained on test images. We end with a general conclusion.

Chapter 01

“Reminder of image restoration concepts”

chapter 1: Reminder of image restoration concepts.

1.1 Introduction:

In this recent time, the image plays a very big role in our life; it helps to represent ideas, embody meanings that we couldn't express with words, furthermore, it helps to remember old memories. Indeed, the fluctuations around the average intensities of the image, inherent either to the acquisition device (camera, amplifiers, quantization, etc.) or to the scene itself (dust, etc.) constituting disturbances which affect the quality of Image and are generally referred to as image noise. The elimination of noise and the production of quality images are therefore of great importance.

One of the major problems encountered in digital imaging concerns the degradation of images. The object of restoration is to reduce, if not eliminate, the distortions introduced by the system used to acquire the image. The principle is to model the noise to find the ideal, non-noisy image from the degraded image.

In this chapter, we are going to introduce basic notions on image processing field, and image restoration by modeling the degradation/restoration process, moreover present the reasons and types of noise and degradation, also some important definitions.

1.2 Definition of image, digital image, pixel:

1.2.1 Image definition:

An image is a concrete or abstract representation of an object, a living being or even a concept. [3]

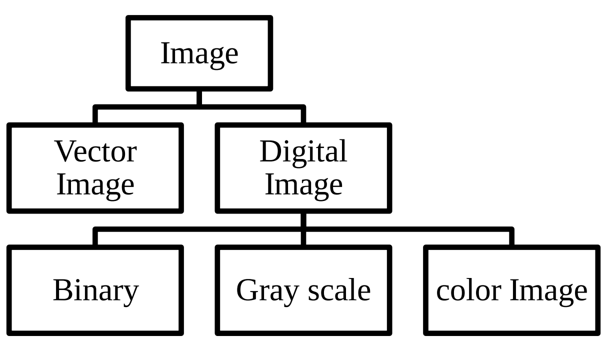


Figure 1.1: Image classifications

1.2.2 Digital image definition:

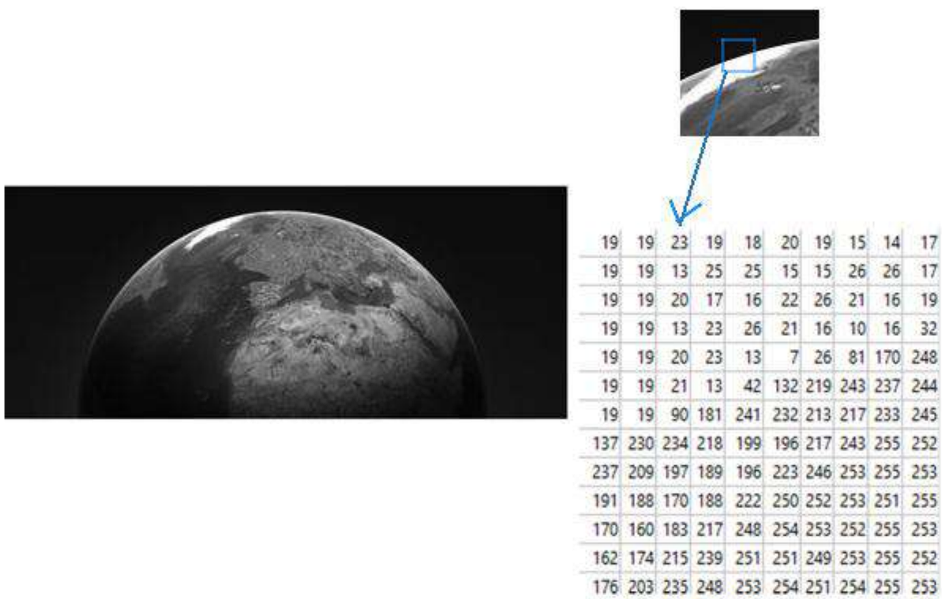
A digital image is a graphic representation based on colored spots (pixels), generally represented on a dotted surface. [3]

A digital image is a matrix of pixels identified by their coordinates (x; y). If it is a color image, a pixel is coded by 3 components (r, g, b) each comprised in the broad sense between 0 and 255, representing respectively the "doses" of red, green and blue which characterize the color of the pixel. If it is a grayscale image, it is coded by a component in the broad sense between 0 and 255, representing the brightness of the pixel. Indeed, each integer representing a gray level is coded on 8 bits. It is therefore between 0 (white) and $2^{(8-1)}$ (black). [4]

1.2.3 Pixel definition:

Pixel is the smallest homogeneous surface constituting a recorded image, defined by the dimensions of the sampling mesh. [3]

Example:



19	19	23	19	18	20	19	15	14	17
19	19	13	25	25	15	15	26	26	17
19	19	20	17	16	22	26	21	16	19
19	19	13	23	26	21	16	10	16	32
19	19	20	23	13	7	26	81	170	248
19	19	21	13	42	132	219	243	237	244
19	19	90	181	241	232	213	217	233	245
137	230	234	218	199	196	217	243	255	252
237	209	197	189	196	223	246	253	255	253
191	188	170	188	222	250	252	253	251	255
170	160	183	217	248	254	253	252	255	253
162	174	215	239	251	251	249	253	255	252
176	203	235	248	253	254	251	254	255	253

Figure 1.2: A digital image representative of the pixel.

1.3 Image restoration:

Image processing is a field that looks after the image treatment by computer, a way of enhancement, optimization, analyzing digital images, by using a set of algorithms and various techniques in order to obtain the desired result; this field have different steps, and among the most important of them is the image restoration.

Image restoration is an objective process aims to improve and fix the distorted /noisy images, trying to return to the clean, original one if it was recorded without deterioration. But if it is recorded with deterioration, then prior knowledge of these phenomena should be used to get the desired result.

Restoration tries to reconstruct by using a priori knowledge of the degradation phenomenon. It deals with getting an optimal estimate of the desired result. Some restoration techniques are best achieved in the spatial domain, while there are some cases where frequency domain techniques are better suited [5].

- **Clarification about the deference between image restoration and image enhancement:**
  - The image restoration is objective whereas the image enhancement is subjective.
  - The enhancement is a process which pursues to improve bad images for looking better while the restoration is a process which pursues to invert known degradation operations which applied to the image.
  - Filters used in the restoration process are different from the filters used in enhancement.

1.4 Modelisation of the degradation/restoration process:

The image degradation/restoration process consists of modeled a degradation function “H” with an additive noise term “$\eta(x, y)$” on an original image “$f(x, y)$” to get a noisy/degraded image “$g(x, y)$”, after that applied restoration techniques which called filters to do the inverse process of the degradation in order to recover the original image and get finally the restored image “$\hat{f}(x, y)$”.

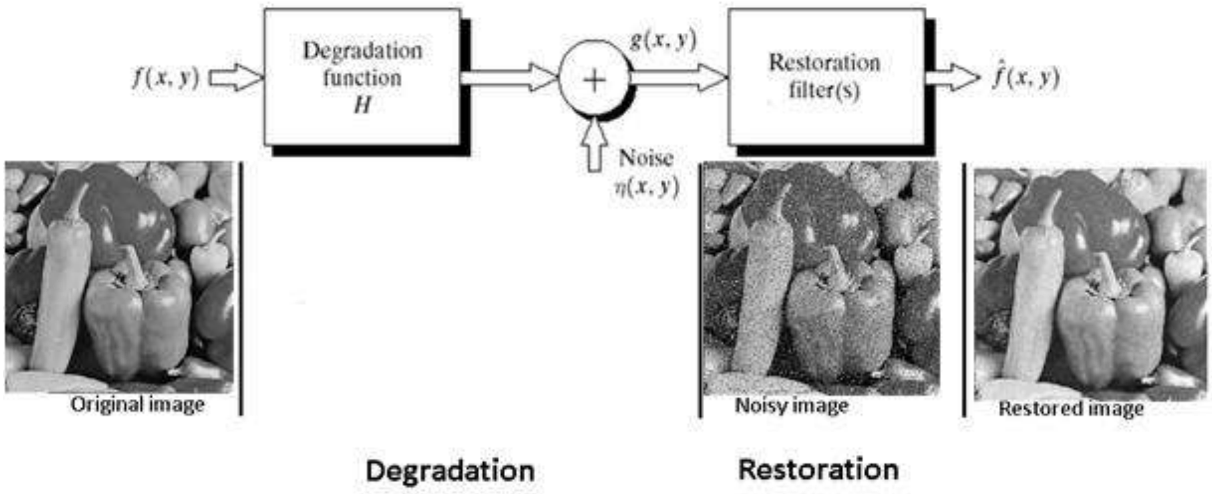


Figure 1.3: Model of the image degradation/restoration process.

Where:

$f(x, y)$: Input image.

$g(x, y)$: Degraded image.

$\eta(x, y)$: Additive noise term.

$\hat{f}(x, y)$: Restored image.

- **Sources and reasons of degradation:**

The degradation (noise) in an image is caused by an external disturbance so it's important to know these disturbances to make the restoration process easier.

a. During transmission:

While transferring an image from one place to another via satellite or wireless transmission, or over the network cable.

b. During acquisition:

While taking the picture, exposure can be done like the blur which can come from an error in the handling of the acquisition material such as a bad focus or camera shake.

c. Imaging sensors:

The degradation can be caused by the bad quality of the sensor or the incorrect use of it or it can be affected by the ambient conditions.

d. **Environment:**

The environment has a major impact on the image as it causes noise to appear in the captured images, embodied in the fall of rain, cloud cover that obscures the light, strong winds, scatted and reflected light, high temperature that can cause the sensor to overheat and thus affects the amount of noise produced.

1.5 Degradation process:

Degradation is losing images there quality because of different reasons. It has several types, among them: invariant linear degradation by translation, additive degradation (noise) ... etc.

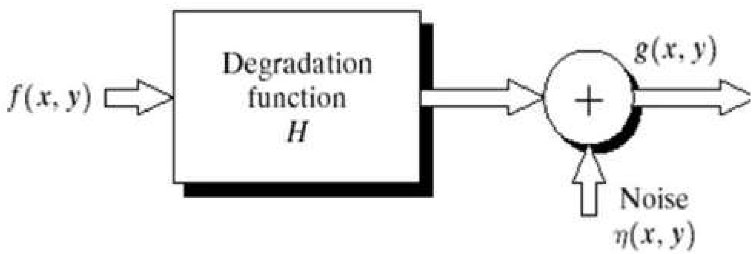


Figure 1.4: Degradation image model.

Degradation image expression:

In spatial domain, the degradation of the original image can be modeled as [6]:

$$g(x,y) = f(x,y) * h(x,y) + \eta(x,y) \quad (1.1)$$

Where:

(x, y): Detached pixel coordinates of the image frame.

$f(x, y)$: Original image.

$g(x, y)$: Degraded image.

h(x, y): Image degradation function.

$\eta(x, y)$: Ad-on noise.

As convolution operation within the spatial domain corresponds to multiplication in the frequency domain, the equation (1.1) can be rewritten as:

$$G(u,v) = F(u,v)H(u,v) + N(u,v) \quad (1.2)$$

1.5.1 Point spread function:

The linear position-invariant function $h(x, y)$ is defined as a point spread function, the PSF is the degree to which an optical system blurs (spreads) a point of light [7]. The point spread function convolved with the original image to return the degraded image.

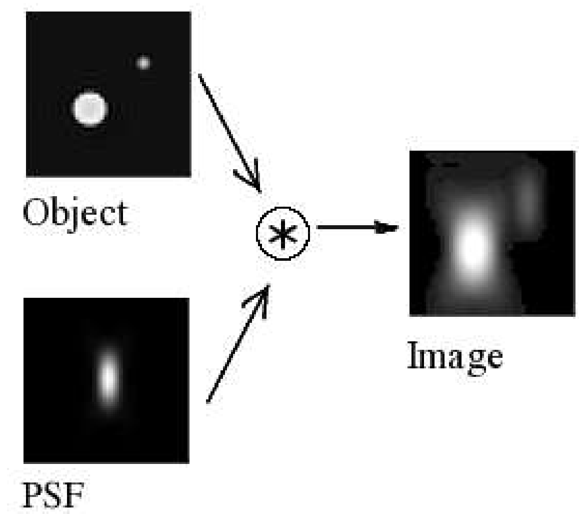


Figure 1.5: Convolution of a point object with the point spread function offers the observed image

1.5.2 Invariant linear degradation by translation:

This type of degradation is with ignoring noise, so degradation is expressed as a convolution:

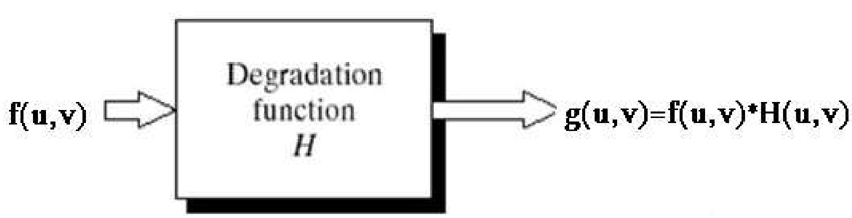


Figure 1.6: Modeling of degradation process without noise

1.5.3 Without ignoring noise:

Noise means random variation of intensity level. Some additional information is added to the pixels and makes the noisy image. The pixel which is shown in the image is not the correct pixel some extraneous value is added to the true pixel value. [8]

The chosen Filter in image restoration depends on noise's type which the image has corrupted by. So it's very important to recognize the noise's types and models.

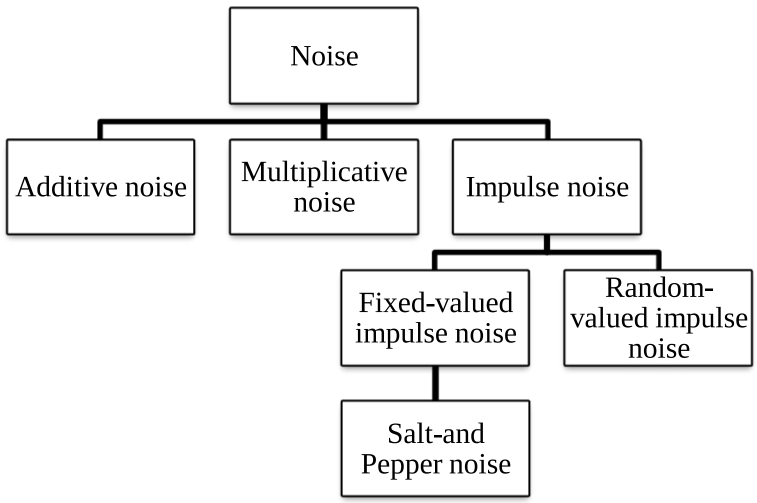


Figure 1.7: Types of noise

a. Additive noise:

Additive noise is easy to model and it can be reduced or removed easily. It is independent of signal. Mathematically, it is described as

$$t(p,q) = o(p,q) + n(p,q) \tag{1.3}$$

Where, t is the observed image with noise, and o is the true signal (image), and n is the noise component. [8]

b. Multiplicative noise:

Whereas multiplicative noise is complex to model and it is difficult to reduce or remove. It is signal dependent. Mathematically, it is described as

$$t(p,q) = o(p,q) * n(p,q) \tag{1.4}$$

Where, t is the observed image with noise, and o is the true signal (image), and n is the noise component. [8]

c. Impulse noise:

Impulse noise is another type of noises which is caused by malfunctioning pixels in camera sensors, faulty memory locations in hardware, or transmission in a noisy channel. Impulse noise can be classified into two types: fixed-valued impulse noise and random-valued impulse noise.

The fixed-valued impulse noise is also called salt-and pepper noise where the grayscale value of a noisy pixel is either minimum or maximum in gray-scale images. The grayscale values of noisy pixels corrupted by random-valued impulse noise are uniformly distributed in the range of [0,255] for gray scale images, [9] in this case impulse noise can be assumed as an additive noise [10], and randomly damages the pixel, at random positions [11].

As example of the impulse noise we have the Salt and pepper noise:

- **Salt and pepper noise:**

Salt and pepper noise appears as dark pixels in bright regions of the image and bright pixels in dark regions of the image.[12] It is generally caused by defect of camera sensor, software failure, or hardware failure in image capturing or transmission. Due to this situation, salt & pepper noise model, only a proportion of all the image pixels are corrupted whereas other pixels are non-noisy. [13]

The salt and pepper noise is also called shot noise, impulse noise or spike noise.

In the salt and pepper noise there are only two possible values exists that is a and b and the probability of each is less than 0.2. If the numbers greater than this numbers the noise will swamp out image. For 8-bit image the typical value is 255 for salt noise and pepper noise is 0. [9]

The probability density function (PDF) of this noise is shown below in equation (1.5), and Figure.1.8. [9]

$$\mathrm{P(z)}=\begin{cases} pa \;\; for\; z=a \\ pb \;\; for\; z=b \\ 0 \;\; otherwise \end{cases} \tag{1.5}$$

If either Pa or Pb is zero, the impulse noise is called unipolar, a and b usually are extreme values because impulse corruption is usually large compared with the strength of the image signal It is the only type of noise that can be distinguished from others visually. [9]

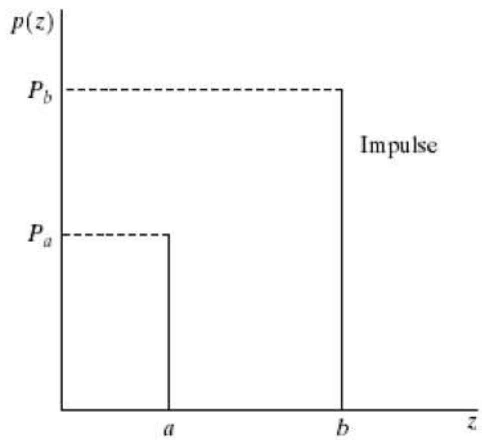
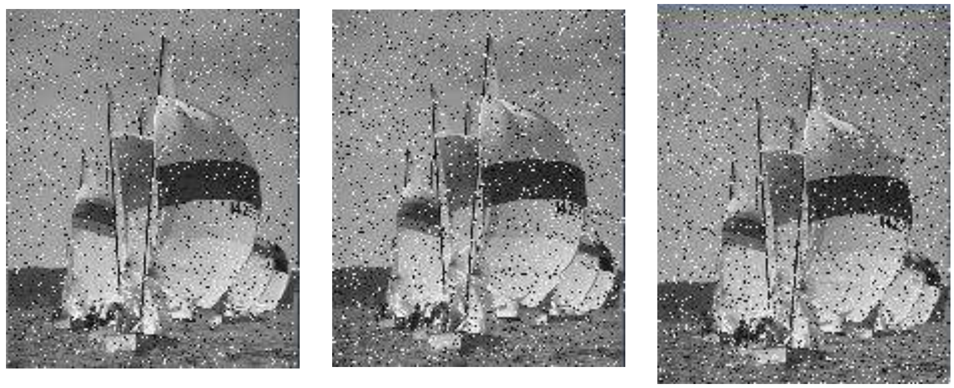


Figure 1.8: PDF of the salt and pepper noise

Salt and pepper noise example:

Adding salt and pepper noise to the image ‘boat.png’ (see Figure.1.9).

Original image. σ=0.02 σ=0.04



σ=0.06 σ=0.08 σ=0.09

Figure 1.9: Examples of salt and pepper image noise "boat.jpg" with different density σ values.

1.6 Restoration process:

Restoration tries to find filters that apply the reverse process on the degraded image in order to find almost the same original image quality.

1.6.1 Restoration filters:

Are the kind of filters that are used for the running of digital noise and estimating the clean and the original image, this restoration filter can take different forms depending on the restoration technique chosen.

The principle of this process is presented by:

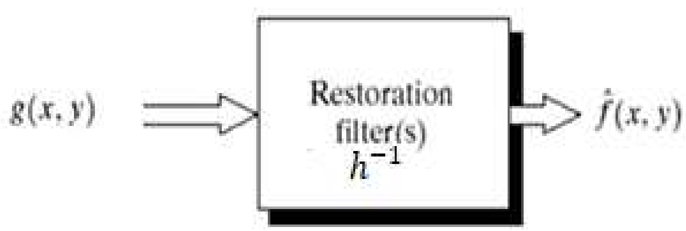


Figure 1.10: Modelisation of restoration process.

h^{-1}: Restore filter.

$\hat{f}$: Restored image.

$g(x, y)$: Degraded image.

1.6.2 Linear/Non-Linear domain:

a. Linear:

To replace a pixel with a linear combination of neighboring pixel values and usually represented by a mask, their methods are more manageable to mathematical analysis, and consequently better understood than the nonlinear ones.

- **Linear filtering principle (Discrete convolution):**

Definition 01: The convolution is a mathematical tool, which makes use of the neighborhood of each pixel, and, therefore, new images are obtained taking into account context (causal) information. This operator is highly time consuming and data dependant. [14]

Definition 02: The convolution is the filtered image of the original image with mask (the filter impulse response).

Why Convolution?

Convolution can accomplish something, which other methods of manipulating images can't realize. Those include blurring, edge detection, noise reduction, sharpening...etc

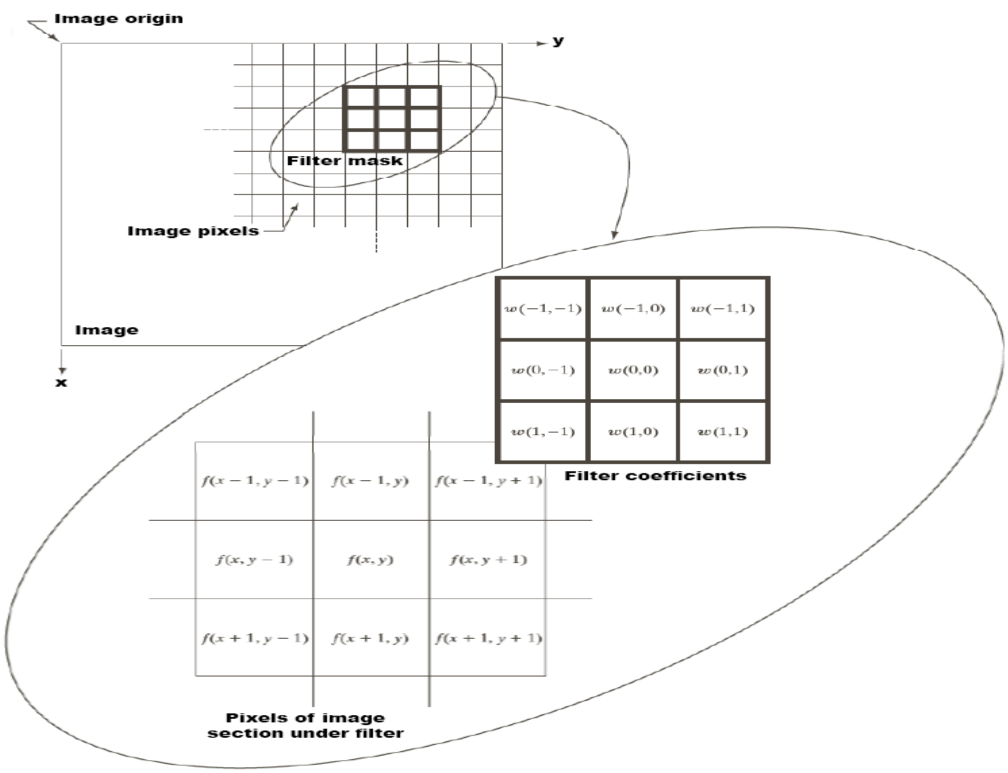


Figure 1.11: The mechanism of linear spatial filtering applying 3×3 filter mask (impulse response array)

Then their convolution $\hat{f}$ is defined as [15]:

$$\hat{f}(x,y) = (h * f)(x,y) = \sum_{i=-x_1}^{x_2} \sum_{j=-y_1}^{y_2} h(i,j) f(x-i, y-j) \quad (1.6)$$

In this expression we can realize:

- $f(x,y)$: The original image.
- $h(i,j)$: The filter impulse response ("mask").

So what happens to borders?

Now we have the problem of the pixels which are on edge and vertices of the image (the mask "exceeds outside"), so what's happening to borders?

It adopts the following three cases:

- Extrapolate by zeros the pixels situated outside the image

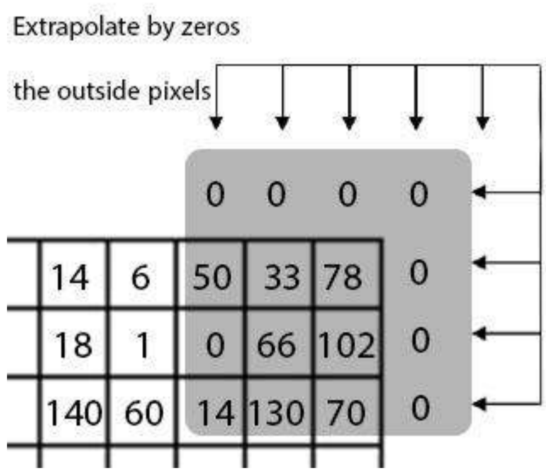


Figure 1.12: Matrix filling in the case of extrapolates by zeros

- Extrapolate through a mirror:

We clone the column or row outward to extend an edge, and we place the extension to the same inner vertex's value to extend a vertex.

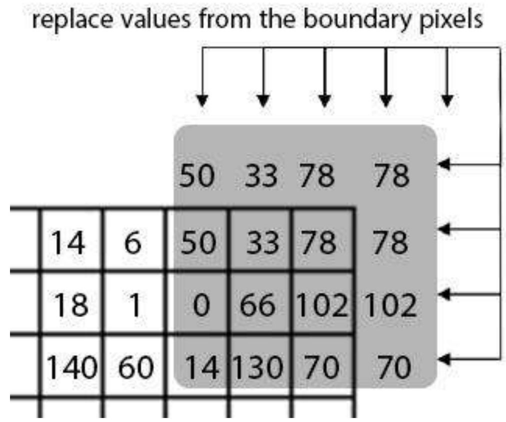


Figure 1.13: Matrix filling in the case of extrapolates through a mirror

- We can also "reduce" the image.

- **Categories of Linear filters:**

Three categories of filters are envisaged in linear filters, which are distinguished by characteristics in the frequency domain:

a. Low-pass filters (LPF): is to preserve the low-frequency components while attenuating the high-frequency, it aims to smooth the image and helps in removing the aliasing effect, also called the "blurring/smoothing" filter, and it's the complement of a high-pass filter.
b. High-pass filters (HPF): is to preserve the high-frequency components while attenuating the low-frequency, it aims to sharpen the image, remove the noise and improves the contrast.
c. Band-pass filters (BPF): keep only power concentrated in a frequency band; they are applying in the case of image restoration. This operation is a derivative of the low pass filter. It consists of eliminating the redundancy of information between the original image and the image obtained by low-pass filtering. Only the difference between the source image and the processed image is kept.[16]

b. Non-Linear mapping:

The calculation doesn't base on linear combinations like the previous one, it's more complicated. It also aims to enhance detail in the darkest regions, on account of the detail in the brighter regions.

There are two aspects of smoothing that non-linear filters are concerned about:

1- Impulse noise: they are more powerful to eliminate the aberrant values than linear Filters.
2- Border integrity: it's able to reduce noise without simultaneously blurring edges.

1.6.3 Filtering in spatial/ frequency domain:

It can broadly be divided in two categories:

a. Spatial domain:

Refers to the image plane itself and deal with images as it is, it methods are based on directly changing the value of the pixels, is preferred when the presence of additive noise only.

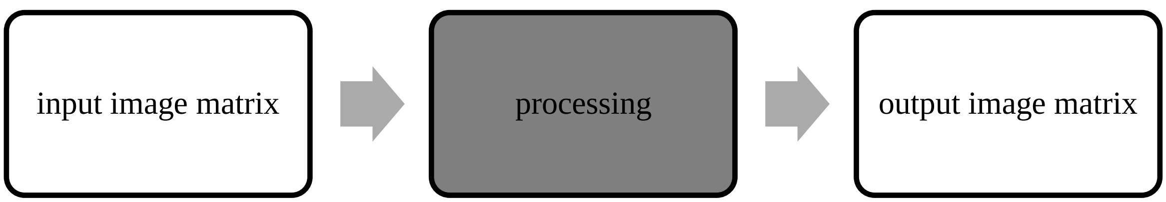


Figure 1.14: Spatial domain filtering operation.

Any image in special domain can be represented in a frequency domain.

b. Frequency domain:

Tells about rate of alteration of intensity per pixel, it's useful for removing periodic noise. Instead of representing an image as an $n * n$ array of pixel values, we can alternatively represent it as the sum of many sine waves of different frequencies, amplitudes and directions. This is referred to as the frequency domain or Fourier representation. The parameters specifying the sine waves are termed Fourier coefficients.

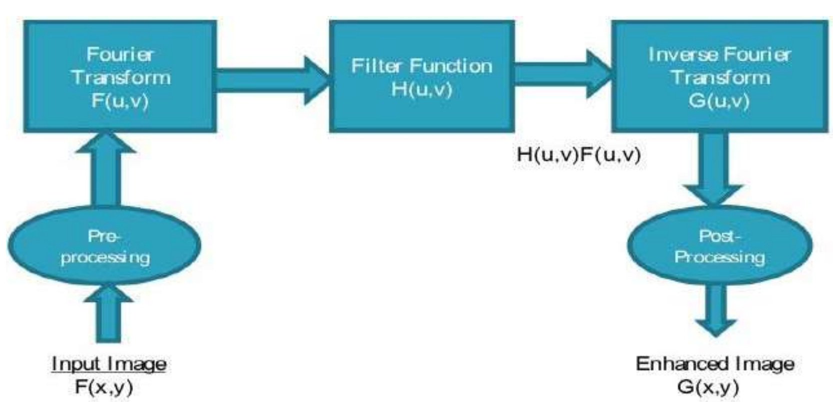


Figure 1.15: Spatial domain filtering operation.

Reason why taking the frequency approach is:

- Extra insight can be gained into how linear filters work by studying them in the frequency domain.
- Some linear filters can be computed more efficiently in the frequency domain, by using the Fast Fourier Transform (FFT).
- New filters can be identified. [17]

1.7 Performance measuring parameters:

In order to evaluate the filters performances and comparing results, different measures of quality appear.

For a given image f ,g, the quality of its restored image, execution evaluation examinations, conserving contours, the reduction of noise, is defined by many measures of quality, and among them we have :

1.7.1 Mean square error (MSE):

The MSE is the combined square error between the original and the encoded image, to find out the average value of the squares error or deviation or find out the difference between the estimator and what is estimated, MSE formula given by equation below: [18]

$$MSE = \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{i=0}^{N-1} |f(i,j) - g(i,j)|^2 \qquad (1.7)$$

Where:

$f(i,j)$: The amplitudes of pixels on the original image.

$g(i,j)$: The amplitudes of pixels on the filtered image.

$M \times N$: The dimension of image.

MSE needs to be as minimum as possible for effective compression.

1.7.2 Peak signal to noise ratio (PSNR):

One of the important performances measuring, when the value of the PSNR is high then the quality of denoised image is good, otherwise considered as bad. The PSNR is regularly communicated as far as the log decibel scale and given by equation: [18]

$$PSNR = 10log_{10}\left(\frac{L^2}{MSE}\right) \tag{1.8}$$

Where:

L: the maximum possible value for a pixel (For the grayscale image$L = 255$).

1.7.3 The Structural Similarity (SSIM) index:

The SSIM is a measure of similarity between two digital images based on structure, instead of a pixel-to-pixel as PSNR does, and this is what SSIM trying to capture (what humans look at images more holistically); is utilized for estimating the closeness between two pictures [18], is defined by:

$$\mathrm{SIMM}(x, y) = \frac{(2\mu_x \mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x{}^2 + \mu_y{}^2 + C_1)(\sigma_x{}^2 + \sigma_y{}^2 + C_2)} \tag{1.9}$$

Where:

- N: Number of pixels
- x_i & y_i: The intensity of a pixel
- μ_x & μ_y : are the measures of the average intensities of x and y calculated for the luminance comparison.

$$(\mu_x = \frac{1}{N}\sum_{i=1}^{N} x_i) \quad 1.10)$$

- σ_x & σ_y: measure of variance for contrast measurement

$$\sigma_x = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)^2} \tag{1.11}$$

- σ_{xy}: is the correlation coefficient

$$\sigma_{xy} = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)(y_i - \mu_y) \quad 1.12)$$

- $C_1 = (K_1L)^2$ & $C_2 = (K_2L)^2$, are added to avoid instability when $\mu_x{}^2 + \mu_y{}^2$ is very near to zero
- L is the dynamics of an image (255 for 8 bits for the grayscale image), where $K_1 \ll 1$, $K_2 \ll 1$ constants.

1.7.4 FSIM (Feature SIMilarity):

Is the similarity feature space between two images I_1 and I_2 on grayscale.

FSIM scores closer to 1 mean higher similarity.

To calculate the FSIM index, we indicate PC_1 and PC_2 the PC maps, and G_1 and G_2 the GM maps obtained from I_1 and I_2. Consists of two stages: [19]

I. Calculate the local similarity map:
   We separate the feature similarity measurement into two components, each for PC orGM.
   a) The similarity measure for $PC_1(x)$ and $PC_2(x)$ are defined as:

$$S_{PC}(x) = \frac{2PC_1(x).PC_2(x) + T_1}{PC_1^2(x) + PC_2^2(x) + T_1} \quad (1.13)$$

Where:

- T_1: A positive constant to grow the stability of S_{pc}.
   b) The similarity measure for $G_1(x)$ and $G_2(x)$ are defined as:

$$S_G(x) = \frac{2G_1(x).G_2(x) + T_2}{PC_1^2(x) + PC_2^2(x) + T_2} \quad (1.14)$$

- T_2: A positive constant depending on GM.

Both T_1 and T_2: will be fixed.

II. Combine the similarity map into a unique similarity score, and we obtain the similarity at $S_L(x)$ for each region x,. We define $S_L(x)$ as:

$$S_L(x) = S_G(x).S_{PC}(x) \quad (1.15)$$

After that weighting the importance of $S_L(x)$in the total similarity between I_1 and I_2:

$$PC_m(x) = \max(PC_1(x), PC_2(x)) \quad (1.16)$$

And according to that the FSIM index of f1 and f2 is defined as:

$$FSIM = \frac{\sum_{x\in\Omega} S_L(x).PC_m(x)}{\sum_{x\in\Omega} PC_m(x)} \quad (1.17)$$

Where Ω means the allover image spatial domain.

1.8 Conclusion:

To this day, images play an important role in the advancement of technology in various fields like television signal processing, medical image, law enforcement, etc... Therefore, it is important to study its development and propose solutions every time to the apparent problems in order to reach higher quality.

In this chapter we have provided the image processing field in general and the degradation and restoration image process in particular, to demonstrate its importance and to clarify each stage of it from the sources of degradation to types of noises until the restoration techniques.

Chapter 02

Simple review on some restoration filters

chapter 2: Simple review on some restoration filters

2.1 Introduction:

The restoration image is a way of mending the degraded image to the nearest real image by eradicating the noise and blur or any haziness. During the image acquisition or transmission or any reason of degradation process, different degradation types occur, reconstructed image restoration techniques have improved by many types of research.
In this chapter, we will discuss the performance analysis of various basics techniques of most useful restoration methods, firstly with the classical methods (Mean, Gaussian, Wiener filters) can be used to restoring the image, secondly will present specific linear and non-linear classes based on PDE, the isotropic diffusion of heat, and the anisotropic diffusion (Perona-Malik), their process depends on the type of noise and fuzziness presents in the image.
Finally we are going to examine the power and weaknesses of each image restoration method.

2.2 Methods used in digital image restoration:

Image restoration techniques focus on applying some inverse procedure to obtain the true image [1], depending on the knowledge of degradation, below some methods of image restoration:

2.2.1 Classical methods

The use of some filtering technique to get the original image back, their algorithms differ depending on the sort of noise present in the image, and below three linear filters:

a. **Linear methods**

- **Mean filter (linear):**

The Mean filter is a linear filter which uses a mask over each pixel in the signal. Each of the components of the pixels which fall under the mask are averaged together to form a single pixel. This filter is also called as average filter. The Mean filter is poor in edge preserving. [20] Has a good result for some specific noise (Gaussian, random, salt and pepper).

Mean filtering is the technique of removing the noise efficiently as the window size increases at the expense of blurring the image. As the window size increases the ability of removing the noise also increases but at the same time the image gets blurred. [1] Is defined by:

$$\hat{f}(x,y) = \frac{1}{MN}\sum_{(u,v)\in S_{xy}} g(u,v) \tag{2.1}$$

Where:

- S_{xy}: Set of coordinates in image window $M \times N$ centered at point(x, y).
- $g(u, v)$: The corrupted image in the area S_{xy}.
- $\frac{1}{MN}$: The convolution mask.

Original image. Noisy image Restored image.

Figure 2.1: Image obtained after applying Mean filter which degraded by the salt and pepper noise.

- **Gaussian filter (linear):**

The Gaussian filter is widely used in the literature as a low-pass filter for image de-noising. It can provide image smoothing and noise reduction, but it blurs edges and details. The Gaussian smoothing filter is efficient for reducing noise drawn from a normal distribution presented as follows:

$$G(x,y,\mu,\sigma)^n = \frac{1}{2\pi\sigma^2}\exp\left[\frac{-(x-\mu)-(y-\mu')^2}{2\sigma^2}\right] \qquad 2.2)$$

Where σ is the spread parameter (the width distribution) and the couple (μ, μ’) are the means (location of the peak) (Figure2.2). [12]

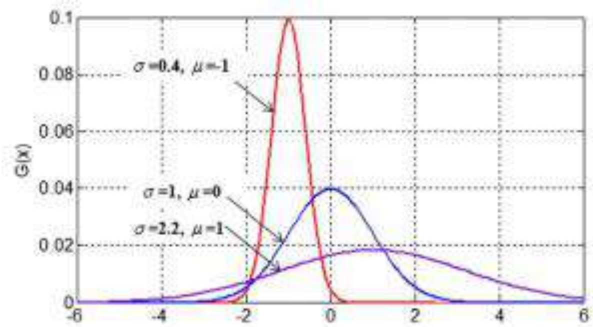


Figure 2.2: Variation of the gaussian functions.

Original image. Noisy image Restored image.

Figure 2.3: Image obtained after applying Gaussian filter which degraded by the salt and pepper noise.

- **Wiener filter (linear):**

Also known as minimum mean square filtering. The Wiener deconvolution was presented by Norbert Wiener, who first proposed the method in 1942, Wiener filtering is one of the earliest and best known approaches to linear image restoration [21], The Wiener filter is the appropriate filter for handling the images which has been degraded by noise as well as by motion blur. [1] It has been used for filtering the observed noisy image. It is used for additive noise, noise spectra and stationary signal from the image [18].

The objective of this technique is to find an estimate $\hat{f}$ of the original image f such that the mean square error between them is minimized. This error measure is given by:

$$e^2 = E\{(f - \hat{f})^2\} \quad (2.3)$$

Where: $E\{.\}$ the expected value of the argument. The method is founded on considering image and noise as random processes and objective is to find an estimate $\hat{f}$ of the uncorrupted image f such that the mean square error between them is minimized. If the noise is zero, then the noise power spectrum vanishes and the wiener filter reduces to the inverse filter. [22]

Original image. Noisy image. Restored image.

Figure 2.4: Image obtained after applying Wiener filter which degraded by the salt and pepper noise.

2.2.2 Methods based on PDE (Partial Differential Equation):

For a long time, an increased request at the application for efficient and effective algorithms for enhancement and restoration. In a few years, the partial differential equations have been appeared to represent a tool that introduction in image processing. The PDEs make it possible to obtain results of good restoration problems and unique solutions. Two methods are widely used of partial differential equations, the isotropic diffusion, and the anisotropic diffusion.

a. Isotropic diffusion of heat:

The isotropic diffusion of heat is the most known linear filtering.

The diffusion process says "Isotropic" if it is applied identically in all directions regardless of the position in the image. So Isotropic diffusion of the Heat equation is effective in eliminating noise, but does not allow preservation of the edges (introduce the blur). (Figure 2.5)

Original image. Noisy image. Restored image.

Figure 2.5: Image obtained after applying Heat equation with i=10, λ=0.2; which degraded by the salt and pepper noise.

The Heat equation by Koenderink [23] expressed in the form of linear parabolic PDE:

$$\begin{cases} \frac{\partial u}{\partial t}(x,y,t) = u_{xx}(x,y,t) + u_{yy}(x,y,t) \\ u(x,y,0) = u_0(x,y) \end{cases} \qquad (2.4)$$

Where:

$u_0(x,y)$: the original image.

$u(x,y,t)$: the restored image after a diffusion time t.

Its form of divergence:

$$\begin{cases} \frac{\partial u}{\partial t}(x,y,t) = div(\nabla(u(x,y,t))) \\ u(x,y,0) = u_0(x,y) \end{cases} \qquad (2.5)$$

Where:

div: The divergence operator.

∇: The gradient operator with respect to the spatial variables x and y.

Rachid DERICHE and Olivier FAUGERAS [24] talked about diffusion isotropic and its relationship with EDP, so they summarized the way it appeared and its characteristics as follows:
The linear parabolic EDP [25] allows isotropic diffusion. This diffusion thus operates identically in all directions and has no privileged direction. For noisy image restoration tasks, this clearly has drawbacks. In fact, in regions of homogeneous intensity, this process will effectively reduce the effect of noise, but in regions with discontinuities in the intensity in gray level, these will also be smoothed and the visual contrast. Of these parts will be appreciably reduced, consequently reducing the qualitative and visual interest of such a process. To deal with this problem, ideas of anisotropic diffusion have been proposed.

b. Anisotropic diffusion "Perona-Malik":

The Perona-Malik (PM) model, first proposed in 1987, is a nonlinear partial diffusion equation that uses an inhomogeneous diffusivity coefficient. It is widely used in image processing for purposes like smoothing, restoration, segmentation, filtering or detecting edges. [26]
Using Gaussian diffusion for image filtration has one great drawback: its homogeneity, which leads to unwanted diffusion of image features like edges. The idea behind the Perona-Malik equation is to modify the Heat equation by adding the diffusivity coefficient depending on space activity in a given part of a picture. [27]
Another method for removing noise is to evolve the image under a smoothing partial differential equation similar to the heat equation which is called anisotropic diffusion. With a spatially constant diffusion coefficient, this is equivalent to the heat equation or linear Gaussian filtering, but with a diffusion coefficient designed to detect edges, the noise can be removed without blurring the edges of the image. [28]
The diffusion process says "Anisotropic" if the diffusion has a preferred orientation and diffuses more in one direction than in the other directions.

For PDE-based methods, Perona-Malik model is considered to be the most classic equation so that it has attracted much attention in recent decades. [27]

2.3 Comparison table:

This table summarizes in a simple way the most important points that distinguish each filter from other, showing their characteristics, strengths, and weaknesses (all the filters motioned in this table.2.1 are defined above)

	Author or creator	Filter	characteristics	Advantages and disadvantages
1995	Makoto Ohki, Anastasios N. Venetsanopoulos	Mean filter	A linear filter, most easiest and intuitive method in implementation.	Poor in edge preserving, removing (Gaussian, random, salt and pepper), As the window size increases the ability of removing noise also increases but at the same time the image gets blurred.
		Gaussian filter	Linear filter has a good performance and also had a separability and isotropy properties.	Effective for smoothing images but it does not respect edges, content, borders, and details.
1942	Norbert Wiener	Weiner filter	A linear filter, its objective's technique is to find an estimate of the original image, it required the PSF.	Good results for image which degraded by noise (additive noise, noise spectra, and the stationary signal from the image), as well as by motion blur.
	Koenderink	Isotropic diffusion of heat	A method based on the Gaussian filter principle, its diffusion can't be controlled because it applying in all	Very effective in noise elimination, contours are damaged; the diffusion is identical in every point of image, smoothed both in homogeneous regions and along the contours.

		directions of the image.	
1987	Pietro Perona and Jitendra Malik	Anisotropic diffusion “Perona-Malik“	A conditional diffusion because it applies either a strong or a weak diffusion depending on the gradient size.	Solution for isotropic diffusion problem, a basis and a reference for many of works in restoration image field, controls diffusion intensity, preserve the contours but for a certain number of iterations so this problem can make an unsatisfactory results

Table 2.1: A comparison table between the mean, Gaussian, Weiner, Heat equation, Perona-Malik filters.

2.4 Conclusion:

So many restoration filters are created in order to resolve image problems, new filters appear time by time, there is filters either based on certain new properties that were not taken into account before; or improve and develop a previous filter, or even use it to establish a new filter. That’s why there is a lot and different filters, each one of them has his own characteristics that make it unique.

In this chapter, we have studies some filters we find it essential to reach our goal, and represent there concept in order to take a general idea of their development; so it start from providing a review of each filter to making a comparative table which includes advantages and disadvantages of them.

Chapter 03

Anisotropic diffusion “Perona-Malik model”

chapter 3: Anisotropic diffusion "Perona-Malik model".

3.1 Introduction:

The anisotropic diffusion is applied for both image restoration and enhancement, this method was proposed by Perona and Malik as a correction to the defect of the isotropic diffusion; that's why PM model was the first and basic invented anisotropic diffusion model, which took a great interest from researchers.

In this chapter, first we introduce the principle of PM model, then discuss how does it filter out the noise which makes use of anisotropic diffusion, and finally how the edge stopping function works to sharp edges and preserving well the fine details.

The objective of this chapter is to explain and clarify the steps it goes this model through in order to restore the image.

3.2 Anisotropic diffusion:

After the isotropic diffusion, the anisotropic method appears as a non-linear diffusion. Pietro Perona and Jitendra Malik were the first to adopt this method, in order to achieve better results in image restoration and enhancement field.

3.2.1 Perona-Malik model:

PM model objective is to eliminate noise and preserve contour at the same time, by applying smoothing on the inter-region areas of the image more than the intra-region areas; using their special equation (Equation.3.1).

$$\begin{cases} \dfrac{\partial u}{\partial t} = div(c(|\nabla u(x,y,t)|)\nabla u(x,y,t)) \\ u(x,y,0) = \theta(x,y) \end{cases} \tag{3.1}$$

Where $u_t = u(x,y,t)$ = image obtained after a diffusion time t, div is the divergence operator and ∇ is the gradient operator with respect to the spatial variables x and y. $|\nabla|$ is the local gradient magnitude. And $c(.)$ is so-called diffusion coefficient or edge stopping function. [24] In the case where the edge stopping function is constant, the isotropic diffusion equation can be found.

We note that Perona-Malik diffusion depends on 02 essential parameters:

- **The gradient magnitude threshold parameter (k):**

Its role is to determine the size of the gradients which will be preserved. [29]

- **Diffusion coefficient / Edge stopping function:**

In Perona-Malik model the diffusion coefficient $c(|\nabla \mathrm{I}|)$ is a function of local image gradient. Its value is inversely proportional to the magnitude of the gradient. Since the magnitude of gradient is weak within uniform or inner regions, diffusion coefficient almost 1, so it acts as heat equation, it smoothens the inner region and removes the noise. Near boundaries the magnitude of the gradient will be strong, thereby diffusion coefficient is almost zero, so the diffusion is stopped across boundaries and it preserves the edges. [24]

The diffusion function c is an edge indicator function that controls the diffusion process by reducing the effect of diffusion near edges and behaves locally as the heat equation. [29]

The edge stopping function c(s) is chosen theoretically satisfying two conditions:

a. One is$\lim_{s\to 0} c(s) = 1$, so that rate of diffusion is high within uniform or inner regions.
b. The other one is$\lim_{s\to \infty} c(s) = 0$, so that the diffusion is totally zero across boundaries.

The important property of edge function is that they should have a zero value or insignificant value for those gradients that corresponds to edges. [24]

The (Table.3.1) follow present different edge stopping functions which are used the most and suggested by different researchers like Perona and Malik, Black et.al, Zhichang Guo et.al, Weickert:

The most famous edge stop functions	Proposed by
$c1(|\nabla \mathrm{I}|) = \exp\left(-\left(\frac{|\nabla \mathrm{I}|}{\mathrm{K}}\right)^2\right)$	Perona and Malik
$c2(|\nabla \mathrm{I}|) = \frac{1}{1+\left(\frac{|\nabla \mathrm{I}|}{\mathrm{K}}\right)^2}$	Perona and Malik
$c3(|\nabla \mathrm{I}|) = \frac{1}{2}\left[1-\left(\frac{|\nabla \mathrm{I}|}{\mathrm{S}}\right)^2\right]^2$; $|\nabla \mathrm{I}| \leq S$ 0 ; otherwise Where $S = K\sqrt{2}$.	Black et.al

$c4(\lvert\nabla I\rvert) = \dfrac{1}{1 + \left(\frac{\lvert\nabla I\rvert}{K}\right)^{\alpha(\lvert\nabla I\rvert)}}$ With: $\alpha(\lvert\nabla I\rvert) = 2 - \dfrac{2}{1+\left(\frac{\lvert\nabla I\rvert}{K}\right)^2}$	Zhichang Guo et.al
$c5(\lvert\nabla I\rvert) = 1 - \exp\left(-3.31488 * \dfrac{K^8}{(\lvert\nabla I\rvert)^8}\right); if\ (\lvert\nabla I\rvert) \neq 0$ 1 ; Otherwise	Weickert

Table 3.1: Some edge stopping functions equations.

Where: $\lvert\nabla I\rvert$ is the threshold or contrast parameter.

3.2.2 ESF's Perona-Malik:

As we see in the (Table.3.1) above, Perona and Malik has suggested two edge stopping functions:

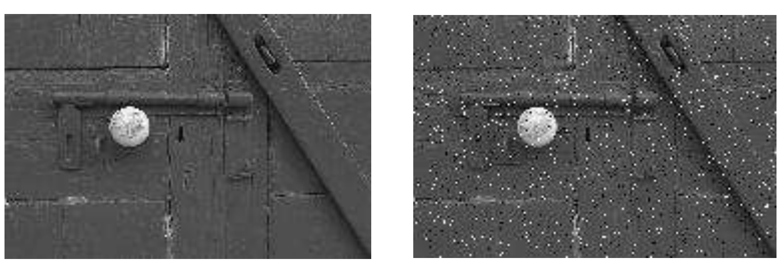
$$g1 = c1(\lvert\nabla I\rvert) = \exp\left(-\left(\frac{\lvert\nabla I\rvert}{K}\right)^2\right) \tag{3.2}$$

Or

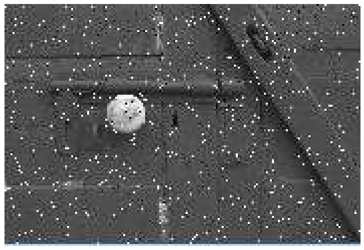
$$g2 = c2(\lvert\nabla I\rvert) = \frac{1}{1 + \left(\frac{\lvert\nabla I\rvert}{K}\right)^2} \tag{3.3}$$

Using "door.jpeg" image (Figure.3.2) we compare the result of each edge stopping function g1 and g2 (Table.3.2) (Table.3.3) by applying the salt-pepper noise on the original image and after that restored them by the Perona-Malik filter.

This comparison use a fixed density σ=0.04 of the salt-pepper noise and different values of threshold parameter k (k=0.2, k=0.6, k= 1, k=2, k=2.5, k=3, k=5), number of iterations i=4, and timestep λ=0.25.

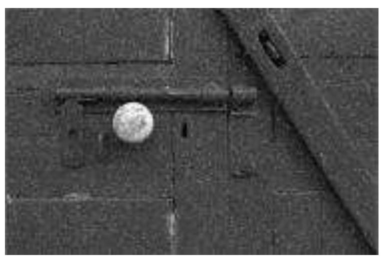


Original image | Noisy image

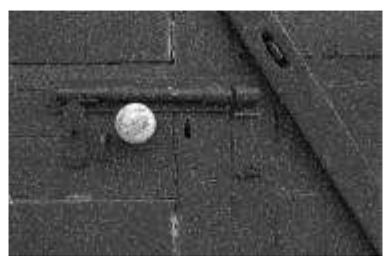


Restored by PM filter using g1 with k=0.2 | Restored by PM filter using g2 with k=0.2

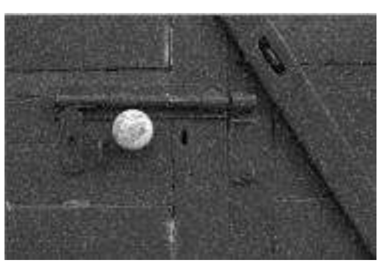
Restored by PM filter using g1 with k=0.6 | Restored by PM filter using g2 with k=0.6



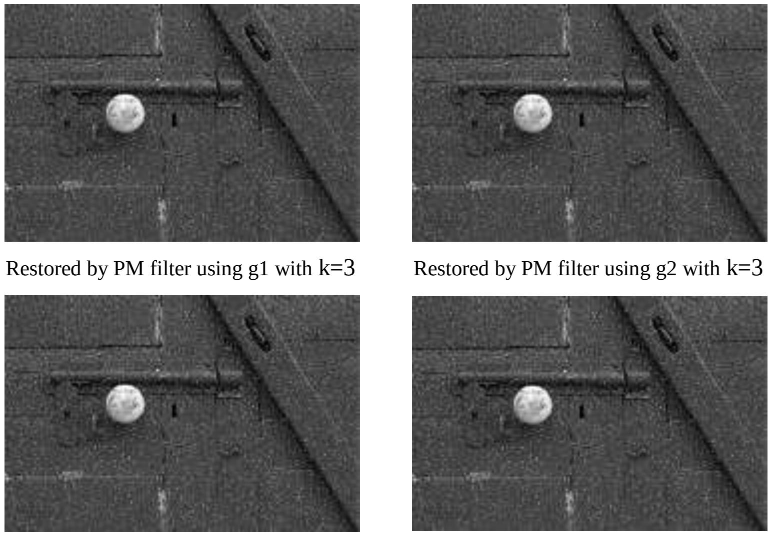
Restored by PM filter using g1 with k=1 | Restored by PM filter using g2 with k=1



Restored by PM filter using g1 with k=2 | Restored by PM filter using g2 with k=2



Restored by PM filter using g1 with k=2.5 | Restored by PM filter using g2 with k=2.5



Restored by PM filter using g1 with k=3 — Restored by PM filter using g2 with k=3

Restored by PM filter using g1 with k=5 — Restored by PM filter using g2 with k=5

Figure 3.1: Comparison between Perona-Malik model edge stopping function g1 and g2 using different values of threshold parameter k, number iterations i, and timestep λ.

		PSNR	SSIM	FSIM
k=0.2	g1	19.57	0.32	0.79
	g2	22.76	0.41	0.81
k=0.6	g1	27.89	0.64	0.85
	g2	28.11	0.66	0.86
.k=1	g1	28.09	0.66	0.86
	g2	28.06	0.65	0.87
k=2	g1	27.62	0.61	0.87
	g2	27.60	0.61	0.87
k=2.5	g1	27.49	0.60	0.87
	g2	27.48	0.60	0.87
k=3	g1	27.41	0.59	0.87
	g2	27.41	0.59	0.87
k=5	g1	27.28	0.58	0.87
	g2	27.28	0.58	0.87

Table 3.2: PSNR, SSIM, and FSIM comparisons for PM model using g1 and g2 edge functions.

At first, by compared the results of PM model using g1 and g2 edge stopping functions at a low values of diffusion (k=0.2, k=0.6) it can be seen that g2 gives better results than g1 even that quality of the image restored is low. As the diffusion value increases, the edge stopping function's results become more convergent until their performances are equal. At a very high quality of the image restored given by PM model we notice from both visual quality and parameters measuring values that g1 gives more satisfied results than g2, which prove that g1 preserves the edges and fine details more than g2 with a good result.

As long as the diffusion value used in the edge stopping function increase, the PSNR and the quality of image decrease, which leads to loss sharp edges and fine details, the two edges stopping functions will have the same effect and give the same result on the noisy image.

Finally, we can include that it's necessary to testing and does the right choice of diffusion value and edge stopping function for better results of Perona-Malik restoration filtering on the noisy image.

3.3 Conclusion:

In this chapter, we have presented an illustrative study of the anisotropic diffusion by Perona-Malik model by explaining its method in a detailed way. We have introduced the PM equation, and describing its basic parameters, and finally we have study the performance of the two PM edge stopping functions which plays an important role in the behavior of this diffusion.

Chapter 04

Experimental results

chapter 4: Experimental results.

4.1 Introduction:

In this last chapter, we present the interface which allows the manipulation of the presented work, and discuss the results of the Perona-Malik method on two images. The objective is to compare the results with the four filters (Mean, Gaussian, Wiener, Heat equation).

The PSNR, as well as the SSIM, FSIM are used as measuring parameters.

4.2 Used tools:

4.2.1 Hardware Tools:

This work was realized by 2 types of computer:

- First type:
  Computer brand "DELL", Intel® Core™ i5-4210 CPU @ 1,70 GHz 1.70 GHz, with 8.00 Go of RAM, operating system 64 bits, Windows 10, and graphics cards (Intel® HD Graphics Family, NIVIDIA GeForce 820M).
- Second type:
  Computer brand "ASUS", Intel® Core™ i5-7200 CPU @ 2.50GHz 2.71 GHz, with 4.00 Go of RAM, operating system 64 bits, Windows 10, and graphics cards (AMD Radeon R5 M420, Intel® HD Graphics 620).

4.2.2 Software tools:

The programming platform used in this work is MATLAB which use the MATLAB language, the version used is R2016a "Release 2016 a (a = first, describing the release order in that year)"; we have realized our work on MATLAB because it is a high performance and an easy-to-use scientific programming language, presents a strong mathematical also numerical support about the implementation of advanced algorithms in image processing and computer vision community.

4.3 Development of the graphical interface using the Matlab GUID toolbox:

The figure below shows the structure of our graphical interface: before and after execution:

Figure 4.1: The graphical interface before execution.

Figure 4.2: The graphical interface after execution.

The creation of this interface aims to perform a comparative study between the restoration methods (Mean, Gaussian, Wiener, Heat equation, PM) methods, also a comparison between the two edges stopping functions PM.

The areas used in this interface are:

I. Basic operations:

- Importing and displaying the original image.
- Converting the color image into a grayscale image, after that displaying the grayscale image.
- Reset the input images and parameters.
- Exit the graphical interface.

II. Noise:

- Displaying the degraded image (we have chosen the salt-and-pepper noise).
- Determine the value of density σ by respecting the condition ($0 < \sigma < 1$), if this condition is not respected the bellow windows will appear :

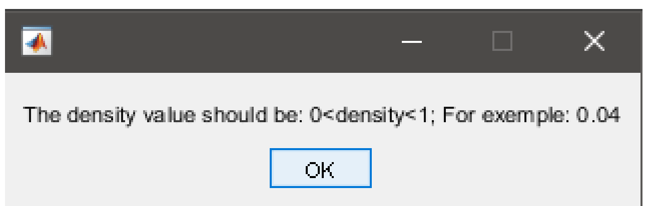
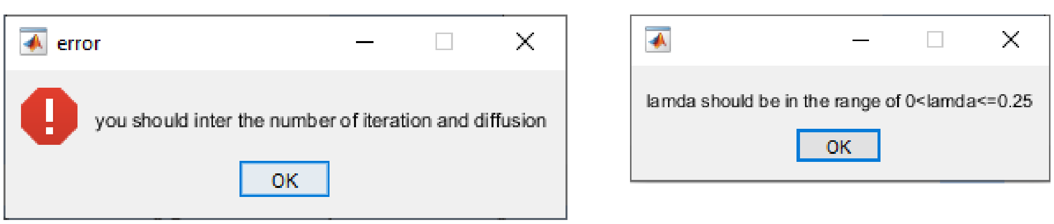


Figure 4.3: The density value condition.

III. Perona-Malik parameters:

- Determine the number of iterations, the diffusion coefficient and the time step. With conditions (the number of iterations i >0, diffusion k>0, the Time step $0 < \lambda \leq 0.25$), if those conditions are not respected the bellow windows will appear :

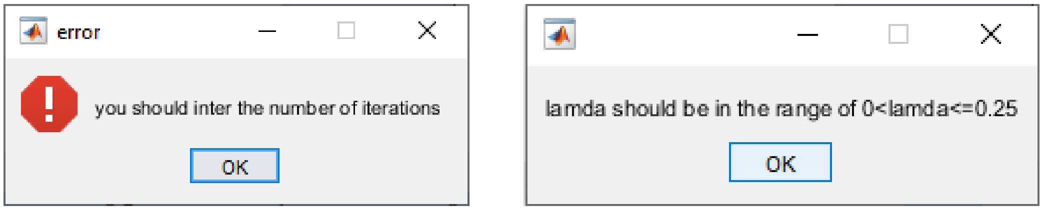


i >0, Diffusion k>0. $0 < \lambda \leq 0.25$

Figure 4.4: Error messages of PM parameters.

IV. Heat equation parameters:

- Determine the number of iterations and the time step. With conditions (The number of iterations i>0, the Time step $0 < \lambda \leq 0.25$), if those conditions are not respected the bellow windows will appear :



i>0 $0 < \lambda \leq 0.25$

Figure 4.5: Error messages of heat equation parameters.

V. Filters:

- Displaying the image restored by the different methods studied in the chapters above, with their measuring parameters.

VI. Comparing between g1 and g2:

- Showing a figure which compared between the edge stopping function g_1, g_2 with their measuring parameters.

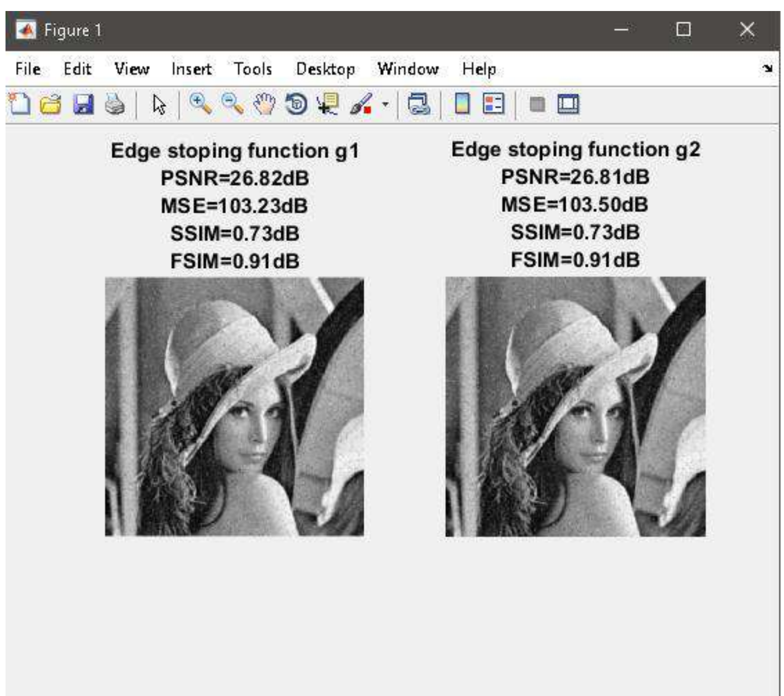


Figure 4.6: Comparison between ESF g1 and g2.

VII. The Help button:

- To have more information about the interface's guide:

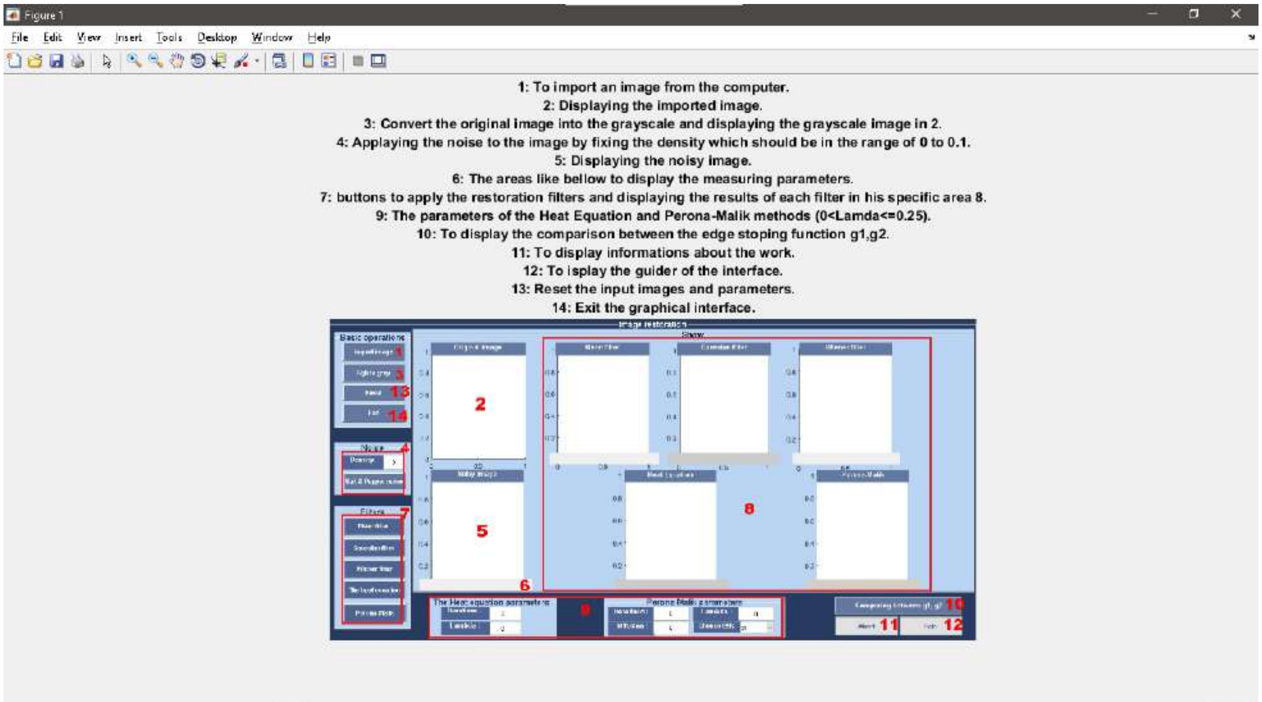


Figure 4.7: The help button.

VIII. The About button:

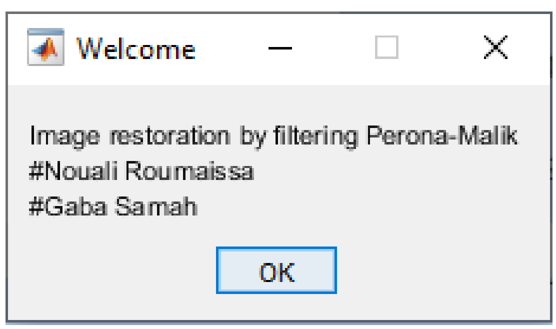


Figure 4.8: The about figure.

4.4 Test images:

In this work, we have worked on three known images of different sizes and types, represented by:

512x512	463x695	512x512	768 x512
Lenna.bmp	Boat.png	Pepper.bmp	door.png

Figure 4.9: Test grayscale images.

4.5 Choice of Perona-Malik method parameters:

The PM method requires a choice of parameters (number of iterations, timestep, and diffusion). By fixing the density of the salt-and-pepper noise (σ=0.04) and by changing the values of parameters (Number of iterations i, timestep λ, diffusion k), comparative tables bellow are presented to prove that those parameters are not easy at all to be chosen, it needs to be chosen carefully according to the noise being treated to get a desirable solution, and show their big impact on image’s quality by using the PSNR, SSIM, FSIM measuring parameters.

IX. Iteration=1:

	i=01		Measuring parameters		
			PSNR On db	SSIM On db	FSIM On db
Noisy image by salt-and-pepper (σ=0.04)	λ=0.1	k=1	21.68	0.43	0.81
		k=3	22.95	0.46	0.82
	λ=0.15	k=1	22.89	0.46	0.82
		k=3	24.71	0.51	0.83
	λ=0.20	k=1	23.96	0.48	0.82
		k=3	**25.57**	**0.53**	**0.83**
	λ=0.25	k=1	24.66	0.48	0.83

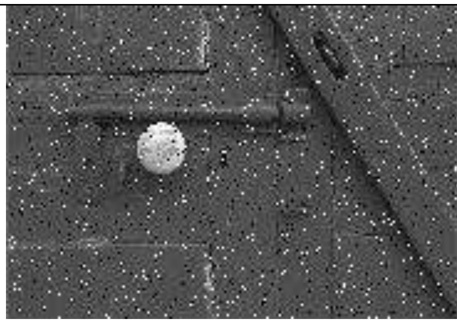
		k=3	24.88	0.47	0.84

Table 4.1: Sensitivity of the parameters (timestep λ, diffusion k, number of iterations i) by fixing i=1 of the PM method for salt-and-pepper noise.

X. Iteration=04:

	i=04		Measuring parameters		
			PSNR On bB	SSIM On dB	FSIM On dB
Noisy image by salt-and-pepper (σ=0.04) 	λ=0.1	k=0.5	22.40	0.44	0.82
		k=1	26.84	0.59	0.84
	λ=0.15	k=3	28.11	0.65	0.86
		k=5	28.12	0.65	0.86
	λ=0.20	k=1	28.15	0.66	0.82
		k=3	**28.27**	**0.67**	**0.87**
	λ=0.25	k=0.5	26.87	0.60	0.84
		k=2	27.64	0.61	0.87

Table 4.2: Sensitivity of the parameters (timestep λ, diffusion k, number of iterations i) by fixing i=4 of the PM method for salt-and-pepper noise.

XI. Iteration=16:

	i=16		Measuring parameters		
			PSNR On dB	SSIM On dB	FSIM On dB
Noisy image by salt-and-pepper (σ=0.04)	**λ=0.1**	**k=1.8**	**28.30**	**0.70**	**0.87**
		k=3.2	28.30	0.70	0.87
	λ=0.15	k=0.5	28.05	0.69	0.86
		k=1	28.09	0.70	0.86

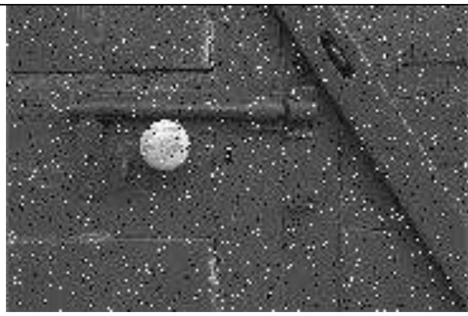
	**λ=0.20**	k=1	27.90	0.70	0.85
		k=0.1	19.06	0.24	0.76
	λ=0.25	k=2	27.58	0.67	0.84
		k=3	27.53	0.66	0.84

Figure 4.10: Sensitivity of the parameters (timestep λ, diffusion k, number of iterations i) by fixing i=16 of the PM method for salt-and-pepper noise.

XII. Iteration=50:

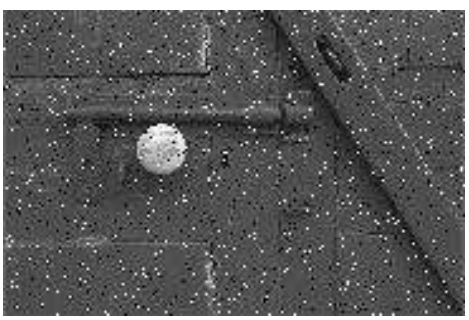
	i=50		Measuring parameters		
			PSNR On dB	SSIM On dB	FSIM On dB
Noisy image by salt-and-pepper (σ=0.04) 	λ=0.1	k=0.5	27.63	0.70	0.83
		k=1	27.61	0.70	0.83
	λ=0.15	k=3	27.29	0.69	0.80
		k=0.2	19.48	0.29	0.70
	λ=0.20	k=2	27.05	0.69	0.78
		k=0.4	27.08	0.69	0.78
	λ=0.25	k=0.4	26.89	0.68	0.76
		k=3	26.80	0.67	0.76

Table 4.3: Sensitivity of the parameters (timestep λ, diffusion k, number of iterations i) by fixing i=50 of the PM method for salt-and-pepper noise.

From the figures and tables above we notice that:

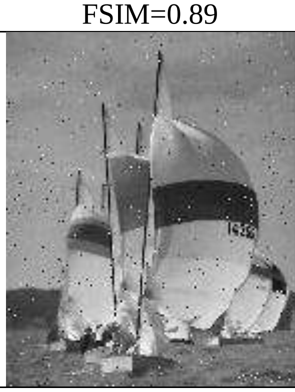
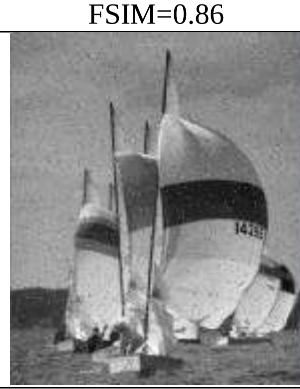
- Perona-Malik model is an iterative process, it’s very sensitive to the number of iterations, no universal criterion for the parameters
- In a specific iteration, the PSNR value is always maximized, which is the ideal time to end the process.
- PM converges towards the mean of the image. The edges "jump" one after another. It must stop at the aspired number of iterations.

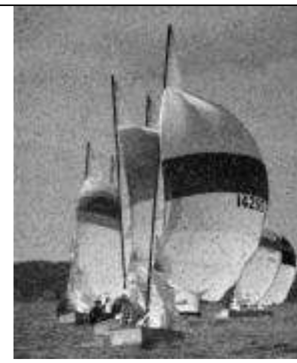
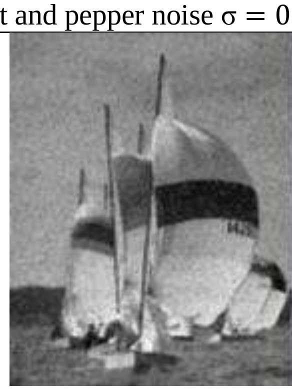
- Choosing the time parameter λ is essential, since overestimating it may blur edges and boundaries while underestimating it may give unfiltered noise artifacts.

4.6 Results of comparison filters:

By changing the density σ of the salt-and-pepper noise values and the images treated with different sizes and types, the restoration methods (Mean, Gaussian, Wiener, Heat equation) are compared with the PM method in the tables below to show their impact on the quality of the images and analyze the performance between each method by using the PSNR, SSIM, FSIM measuring parameters:

- **The image: 'Boat.png':**

		
(a) Noisy image PSNR=22.62 SSIM=0.56 FSIM=0.89	(b) Mean filter PSNR=25.09 SSIM=0.74 FSIM=0.86	(c) Gaussian filter PSNR=5.91 SSIM=0.08 FSIM=0.38
		
(d) Wiener filter PSNR=26.48 SSIM=0.68 FSIM=0.90	(e) Heat equation method (iteration=4, λ=0.25) PSNR=26.82 SSIM=0.73 FSIM=0.90	(f) PM method (iteration=4, k=0.8, λ=0.25) PSNR=27.70 SSIM=0.79 FSIM=0.91
Salt and pepper noise $\sigma = 0.02$		

		
(a) Noisy image PSNR=19.6 SSIM=0.36 FSIM=0.81	(b) Mean filter PSNR=24.87 SSIM=0.71 FSIM=0.85	(c) The Gaussian PSNR=5.87 SSIM=0.07 FSIM=0.36
		
(d) The Wiener filter PSNR=24.8 SSIM=0.60 FSIM=0.86	(e) The Heat Equation (iteration=4, λ=0.25) PSNR=25.96 SSIM=0.64 FSIM=0.87	(f) The PM filter (iteration=4, k=0.8, λ=0.25) PSNR=26.97 SSIM=0.73 FSIM=0.87
Salt and pepper noise $\sigma = 0.04$		
		
(a) Noisy image PSNR=17.88 SSIM=0.26 FSIM=0.75	(b) Mean filter PSNR=24.68 SSIM=0.69 FSIM=0.84	(c) The Gaussian PSNR=5.98 SSIM=0.06 FSIM=0.34

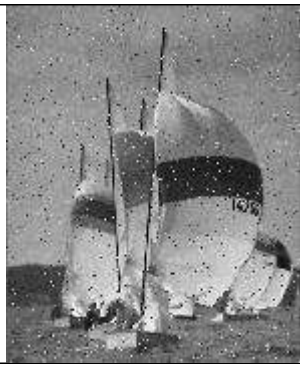
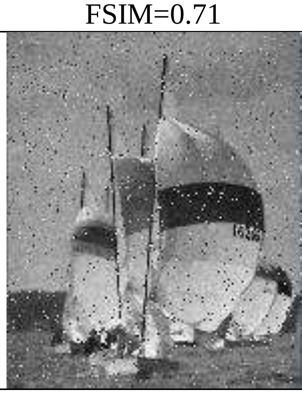
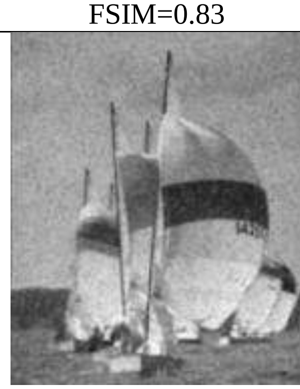
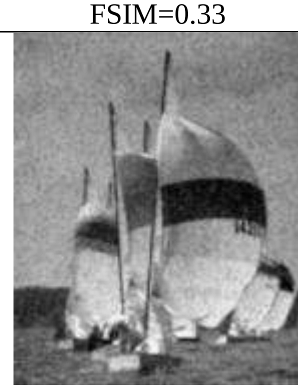
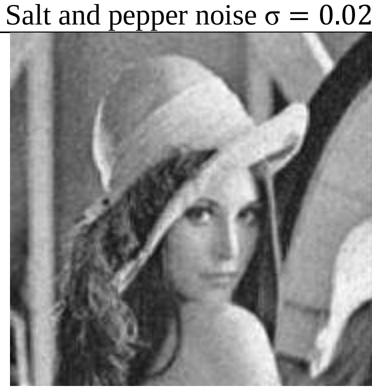
		
(d) The Wiener filter PSNR=23.99 SSIM=0.54 FSIM=0.84	(e) The Heat Equation (iteration=10, λ=0.24) PSNR=24.87 SSIM=0.72 FSIM=0.85	PM(iteration=10, k=1.5, λ=0.24) PSNR=25.32 SSIM=0.73 FSIM=0.85
Salt and pepper noise $\sigma = 0.06$		
		
(a) Noisy image PSNR=16.57 SSIM=0.19 FSIM=0.71	(b) Mean filter PSNR=24.45 SSIM=0.66 FSIM=0.83	(c) The Gaussian PSNR=5.65 SSIM=0.05 FSIM=0.33
		
(d) The Wiener filter PSNR=23.32 SSIM=0.5 FSIM=0.81	(e) The Heat Equation (iteration=10, λ=0.24) PSNR=24.63 SSIM=0.7 FSIM=0.83	(f) PM (iteration=10, k=1.5, λ=0.24) PSNR=25.04 SSIM=0.71 FSIM=0.84
Salt and pepper noise $\sigma = 0.08$		

Table 4.4: Boat.png image restoration by: Mean filter, Gaussian filter, Wiener filter, Heat equation method, and Perona-Malik method in different density sizes of salt and pepper noise.

- **The image: 'Lenna.bmp':**

<table>
<tr><td></td><td></td><td></td></tr>
<tr><td>(a) Noisy image
PSNR=22.38 SSIM=0.60
FSIM=0.91</td><td>(b) Mean filter
PSNR=24.56 SSIM=0.73
FSIM=0.90</td><td>(c) Gaussian filter
PSNR=5.62 SSIM=0.07
FSIM=0.35</td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td>(d) Wiener filter
PSNR=25.60 SSIM=0.67
FSIM=0.92</td><td>(e) Heat equation method
(iteration=4, λ=0.25)
PSNR=25.81 SSIM=0.73
FSIM=0.92</td><td>(f) PM method
(iteration=4, k=0.8, λ=0.25)
PSNR=27.79 SSIM=0.78
FSIM=0.93</td></tr>
<tr><td colspan="3">Salt and pepper noise $\sigma = 0.02$</td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td>(a) Noisy image
PSNR=19.41 SSIM=0.34
FSIM=0.84</td><td>(b) Mean filter
PSNR=24.38 SSIM=0.71
FSIM=0.89</td><td>(c) The Gaussian
PSNR=5.75 SSIM=0.06
FSIM=0.33</td></tr>
</table>

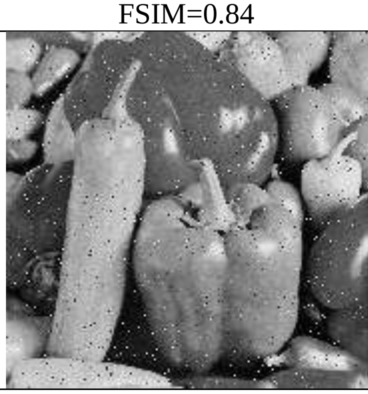
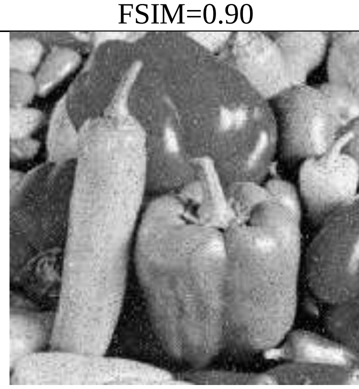
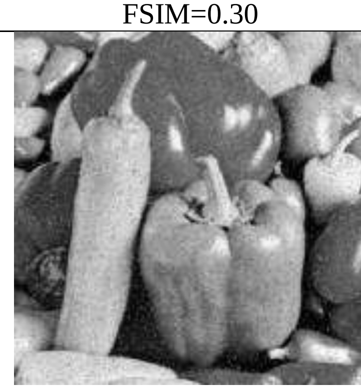
		
(d) The Wiener filter PSNR=23.92 SSIM=0.59 FSIM=0.89	(e) The Heat Equation (iteration=4, λ=0.25) PSNR=24.88 SSIM=0.65 FSIM=0.90	(f) The PM filter (iteration=4, k=0.8, λ=0.25) PSNR=26.79 SSIM=0.73 FSIM=0.91
Salt and pepper noise $\sigma = 0.04$		
		
(a) Noisy image PSNR=17.64 SSIM=0.29 FSIM=0.79	(b) Mean filter PSNR=23.94 SSIM=0.69 FSIM=0.88	(c) The Gaussian PSNR=5.66 SSIM=0.06 FSIM=0.32
		
(d) The Wiener filter PSNR=22.96 SSIM=0.55 FSIM=0.87	(e) The Heat Equation (iteration=10, λ=0.24) PSNR=23.23 SSIM=0.71 FSIM=0.88	(f) PM (iteration=10, k=1.5, λ=0.24) PSNR=25.00 SSIM=0.72 FSIM=0.90
Salt and pepper noise $\sigma = 0.06$		

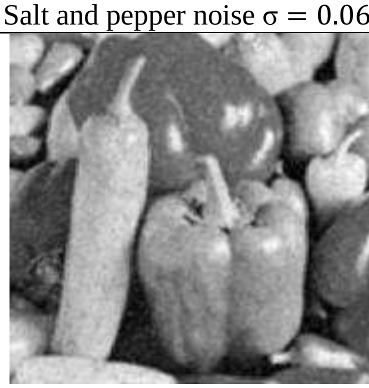
		
(a) Noisy image PSNR=16.43 SSIM=0.22 FSIM=0.75	(b) Mean filter PSNR=23.61 SSIM=0.67 FSIM=0.87	(c) The Gaussian PSNR=5.63 SSIM=0.05 FSIM=0.31
		
(d) The Wiener filter PSNR=16.43 SSIM=0.22 FSIM=0.75	(e) The Heat Equation (iteration=10, λ=0.24) PSNR=22.94 SSIM=0.69 FSIM=0.87	(f) PM (iteration=10, k=1.5, λ=0.24) PSNR=24.06 SSIM=0.70 FSIM=0.89
Salt and pepper noise $\sigma = 0.08$		

Table 4.5: Lenna.bmp image restoration by: Mean filter, Gaussian filter, Wiener filter, Heat equation method, and Perona-Malik method in different density sizes of salt and pepper noise.

- **The image: ‘pepper.png’:**

		
(a) Noisy image PSNR=22.16 SSIM=0.59	(b) Mean filter PSNR=25.02 SSIM=0.76	(c) Gaussian filter PSNR=5.62 SSIM=0.07

FSIM=0.91	FSIM=0.91	FSIM=0.31
		
(d) Wiener filter PSNR=25.55 SSIM=0.66 FSIM=0.93	(e) Heat equation method (iteration=4, λ=0.22) PSNR=27.08 SSIM=0.79 FSIM=0.93	(f) PM method (iteration=4, k=1.3, λ=0.22) PSNR=27.90 SSIM=0.80 FSIM=0.94
Salt and pepper noise $\sigma = 0.02$		
		
(a) Noisy image PSNR=19.27 SSIM=0.39 FSIM=0.84	(b) Mean filter PSNR=24.67 SSIM=0.73 FSIM=0.90	(c) The Gaussian PSNR=5.73 SSIM=0.06 FSIM=0.30
		
(d) The Wiener filter PSNR=23.87 SSIM=0.59 FSIM=0.90	(e) The Heat Equation (iteration=4, λ=0.25) PSNR=25.30 SSIM=0.67 FSIM=0.91	(f) The PM filter (iteration=4, k=0.8, λ=0.25) PSNR=26.74 SSIM=0.75 FSIM=0.91
Salt and pepper noise $\sigma = 0.04$		

		
(a) Noisy image PSNR=17.64 SSIM=0.29 FSIM=0.79	(b) Mean filter PSNR=23.94 SSIM=0.69 FSIM=0.88	(c) The Gaussian PSNR=5.66 SSIM=0.06 FSIM=0.32
		
(d) The Wiener filter PSNR=22.96 SSIM=0.55 FSIM=0.87	(e) The Heat Equation (iteration=10, λ=0.24) PSNR=23.23 SSIM=0.71 FSIM=0.88	(f) PM (iteration=10, k=1.5, λ=0.24) PSNR=25.00 SSIM=0.72 FSIM=0.90
Salt and pepper noise $\sigma = 0.06$		
		
(a) Noisy image PSNR=16.29 SSIM=0.21 FSIM=0.75	(b) Mean filter PSNR=23.92 SSIM=0.69 FSIM=0.88	(c) The Gaussian PSNR=5.73 SSIM=0.05 FSIM=0.29

		
(d) The Wiener filter PSNR=22.19 SSIM=0.51 FSIM=0.85	(e) The Heat Equation (iteration=10, λ=0.24) PSNR=23.45 SSIM=0.71 FSIM=0.88	(f) PM (iteration=10, k=0.9, λ=0.24) PSNR=24.68 SSIM=0.73 FSIM=0.89
Salt and pepper noise σ = 0.08		

Table 4.6: Pepper.bmp image restoration by: Mean filter, Gaussian filter, Wiener filter, Heat equation method, and Perona-Malik method.

Image	Noise density	Measuring parameters		Restoration filters				
			Noisy	Mean	Gaussian	Wiener	Heat equation	PM
Boat.png	σ=0.02	PSNR	22.62	25.09	5.91	26.48	26.82	27.7
		SSIM	0.56	0.74	0.08	0.68	0.73	0.79
		FSIM	0.89	0.86	0.38	0.90	0.90	0.91
	σ=0.04	PSNR	19.6	24.87	5.87	24.8	25.96	26.97
		SSIM	0.36	0.71	0.07	0.60	0.64	0.73
		FSIM	0.81	0.85	0.36	0.86	0.87	0.87
	σ=0.06	PSNR	17.88	24.68	5.98	23.99	24.87	25.32
		SSIM	0.26	0.69	0.06	0.54	0.72	0.73
		FSIM	0.75	0.84	0.34	0.84	0.85	0.85
	σ=0.08	PSNR	16.57	24.45	5.65	23.32	24.63	25.04
		SSIM	0.19	0.66	0.05	0.5	0.7	0.71
		FSIM	0.71	0.83	0.33	0.81	0.83	0.84
lenna.bmp	σ=0.02	PSNR	22.38	24.56	5.62	25.60	25.81	27.79
		SSIM	0.60	0.73	0.07	0.67	0.73	0.78
		FSIM	0.91	0.90	0.35	0.92	0.92	0.93
	σ=0.04	PSNR	19.41	24.38	5.75	23.92	24.88	26.79
		SSIM	0.34	0.71	0.06	0.59	0.65	0.73
		FSIM	0.84	0.89	0.33	0.89	0.90	0.91
	σ=0.06	PSNR	17.64	23.94	5.66	22.96	23.23	25
		SSIM	0.29	0.69	0.06	0.55	0.71	0.72
		FSIM	0.79	0.88	0.32	0.87	0.88	0.90
	σ=0.08	PSNR	16.43	23.61	5.63	16.43	22.94	24.06
		SSIM	0.22	0.67	0.05	0.22	0.69	0.70

		FSIM	0.75	0.87	0.31	0.75	0.87	0.89
Pepper.bmp	σ=0.02	PSNR	22.16	25.02	5.62	25.55	27.08	27.90
		SSIM	0.59	0.76	0.07	0.66	0.79	0.80
		FSIM	0.91	0.91	0.31	0.93	0.93	0.94
	σ=0.04	PSNR	19.27	24.67	5.73	23.87	25.30	26.74
		SSIM	0.39	0.73	0.06	0.59	0.67	0.75
		FSIM	0.84	0.90	0.30	0.90	0.91	0.91
	σ=0.06	PSNR	17.64	23.94	5.66	22.96	23.23	25.00
		SSIM	0.29	0.69	0.06	0.55	0.71	0.72
		FSIM	0.79	0.88	0.32	0.87	0.88	0.90
	σ=0.08	PSNR	16.29	23.92	5.73	22.19	23.45	24.68
		SSIM	0.21	0.69	0.05	0.51	0.71	0.73
		FSIM	0.75	0.88	0.29	0.85	0.88	0.89

Table 4.7: Comparison of performance parameters of the noisy images after restoration.

From the figures and tables above we notice that:

- The results show that the PM method has a higher PSNR, SSIM, FSIM compared with the other methods of restoration.
- The PM is a great filter that preserves the edges of the image however it has some drawbacks; PM needs precise parameters to obtain good results.
- The higher the density σ of noise present in the image, the lower the ideal performance of the PM method. So the PM method returns the best results in the presence of low noise densities.

4.7 Conclusion:

In this chapter, we conducted a comparative study between PM model and the other filters (Mean, Gaussian, Weiner, Heat equation) mentioned in our work previously (Chapter 02) depending on visual quality of the filtered images and the values of measuring parameters (PSNR, SSIM, FSIM) in order to discover the conditions and cases in which PM model provides the best results; we have illustrated this work on a graphical interface.

On the other hand, we have explained the points and deficiencies that this filter suffers from, and the reasons for them as well.

Finally, due to the satisfying results that this filter provides when it is in its optimum conditions, it makes researchers want to develop it to produce better results every time.

General conclusion

General conclusion:

In this thesis, we have presented a study in the field of image processing and image restoration in particular about the PM model based on the EDPs. The objective of this study is to illustrate in detail the PM model and clarify their strengths and weaknesses, how it works, and improve the good quality of the image restored by it; for that we have presented several filters (Mean, Gaussian, Weiner, Heat equation) in order to make a comparison between those methods of image restoration filtering and our model studied using different measuring parameters (PSNR, SSIM, FSIM). The result shows that PM model is more efficient under specific conditions so it's important to choose correctly the parameters that this model depends on.

We have presented our work on a graphical interface implemented with Matlab to see the results and difference between PM model and the other filters with a better way; also this interface helps to see the difference between the results of the used edge stopping functions.

References

References:

[1]: Asha Nathan and Varun Kumar M, “Comparison of Image Restoration and Denoising Techniques”, December 2013.

[2]: Pietro Perona and Jitendra Malik, Scale-Space and Edge Detection Using Anisotropic Diffusion, IEEE transaction on Pattern Analysis and Machine Intelligence, July 1990, VOL. 12, Nov2.

[3]: Dictionnaire d’informatique - Larousse.

[4]: Cours de traitement d’image Master EDPs et Application, Université de M’sila. BEN-HAMIDOUCHE.N,2019.

[5]: Dhananjay K. Theckedath, Digital Image Processing (Using MATLAB Codes), Second Revised Edition: July 2013.

[6]: D. Kundur and D. Hatzinakos, "A novel blind deconvolution scheme for image restoration using recursive filtering," IEEE Transactions on Signal Processing, vol. 46, pp. 375-390, 1998.

[7]: Aizenberg I., Bregin T., Butakoff C., Karnaukhov V., Merzlyakov N. and Milukova O., "Type of Blur and Blur Parameters Identification Using Neural Network and Its Application to Image Restoration". In: J.R. Dorronsoro (ed.) Lecture Notes in Computer Science, Vol. 2415, Springer-Verlag, Berlin, Heidelberg, New York (2002) 1231-1236.

[8]: Alenrex Maity and Rishav Chatterjee, “Article in Computer Vision Graphics and Image Processing”, School of Computer Science Engineering, KIIT University, Odisha, India, March 2018.

[9]: Mr. Mandar D. Sontakke, Dr. Mrs. Meghana S. Kulkarni; “Different types of noises in images and noise removing technique”, Internationnal Journal of Advanced Technology in Engineering and Science, Volume No.03, Issue No. 01, January 2015 ISSN (online): 2348 – 7550.

[10]: E. Abreu, M. Lightstone, S. K. Mitra, and K. Arakawa, “A newefficient approach for the removal of impulse noise from highly Corrupted images,” *IEEE Transactions on Image Processing*, vol. 5, no. 6, pp. 1012-1025, Jun 1996.

[11]: H. Ibrahim, T. F. Ng, and S. H. Teoh, “An efficient implementation of switching median filter with boundary discriminative noise detection for image corrupted by impulse noise,” *Scientific Research and Essays (SRE),* vol. 6, no. 26, pp. 5523-5533, November 2011.

[12]: Seddik Hassene, “A new family of Gaussian filters with adaptive lobe location and smoothing strength for efficient image restoration”, EURASIP Journal on Advances in Signal Processing 2014.

[13]: W. C. M. Van Beers and J. P. C. Kleijnen “Kriging for interpolation in random simulation”, Journal of the Operational Research Society, vol. 54, pp. 255–262, 2003.

[14]: Jose M. Palomares, “General Logarithmic Image Processing Convolution”, IEEE TRANSACTIONS ON IMAGE PROCESSING, NOVEMBER 2006.

[15]: Ramesh Jain, Rangachar Kasturi, Brian G. Schunck, “MACHINE VISION”, book, 1995.

[16]: Développement d’algorithmes de tatouage d’images basés sur la SVD et les transformées Discrètes, université Farhat Abbas-Setif, 2014.

[17]: Woon Siong Gan, “Signal Processing and Image Processing for Acoustical Imaging”, Book, 04 April 2020.

[18]: Reeturaj Mishra, Neetu Mittal, Sunil Kumar Khatri, “Digital Image Restoration using Image Filtering Techniques”, International Conference on Automation, Computational and Technology Management (ICACTM) Amity University, 2019.

[19]: Lin Zhanga, Student Member, IEEE, Lei Zhanga,1, Member, IEEE, Xuanqin Moub, Member, IEEE, and David Zhanga, Fellow, IEEE, “FSIM: A Feature Similarity Index for Image Quality Assessment”.

[20]: Patidar, Pawan, and al. "Image De-noising by Various Filters for Different Noise", International Journal of Computer Applications 9.4 (2010): 45-50.

[21]: Reginald L. Lagendijk and Jan Biemond, “Basic methods for image restoration and Identification”, Lagendijk- Biemond, February, 1999.

[22]: Madri Thakur and Shilpa Datar, “Image Restoration Based On Deconvolution by Richardson Lucy Algorithm”, International Journal of Engineering Trends and Technology (IJETT), Aug 2014.

[23]: J .J. Koenderink. The structure of images. Biol. Cybern, 50:363-370, 1984.

[24]: Rachid DERICHE and Olivier FAUGERAS “PDE's in Image Processing and Computer Vision”

[25]: INRIA Sophia-Antipolis 2004, Route des Lucioles, BP 93 - 06902 Sophia-Antipolis Cedex France.

[26]: Maciej Wielgus, “Perona-Malik equation and its numerical properties”, Article, December 2014.

[27]: Zhiyi Ruan, Youjian Shen and Fengling Liu, “Modified Numerical Scheme for Perona-Malik Model in Image Restoration”, College of Mathematics and Statistics, Hainan Normal University, China.

[28]: Mohd Awais Farooque, Jayant S.Rohankar, “Survey on various noises and techniques for denoising the color image”, ISSN 2319 – 4847 volume 2, Issue 11, November 2013, International Journal of Application or Innovation in Engineering & Management (IJAIEM).

[29]: V Kamalaveni, R Anitha Rajalakshmi, K A Narayanankuttyb; “Image denoising using variations of Perona-Malik with different Edge Stopping Function”; Second International Symposium on Computer Vision and the Internet (VisionNet’15).