

NUMERICAL COMPARISON OF DIFFERENT IMAGING ALGORITHMS.

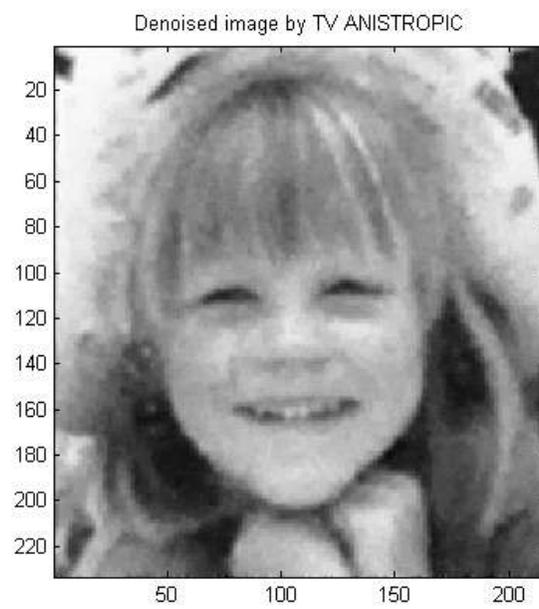
Soulef BOUGUEROUA¹ and —Noureddine 2,1 DAILI¹

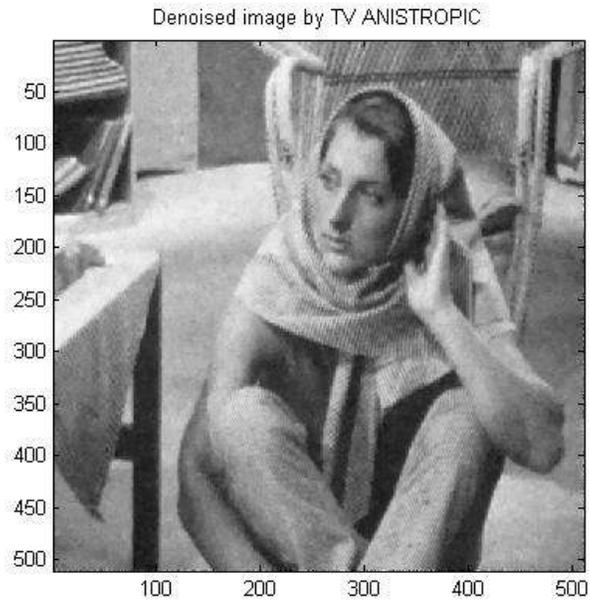
¹Universite Ferhat Abbas Setif 1 Faculte des Sciences

May 26, 2023

Abstract

Image processing is the set of operations performed to extract “information” from the image. An interesting problem in digital image processing is the restoration of degraded images. It often happens that the resulting image is different from the expected image. Our problem will therefore be to recover an image close to the original image from a poor quality image (that has been skewed by Gaussian and additive noise). There are a lot of algorithms on how we can improve the broken image in better quality. We present in this paper our numerical results obtained with the models of Tichonov regularization, ROF, Vese Osher, anisotropic and isotropic TV denoising algorithms.





Original image



Noisy image

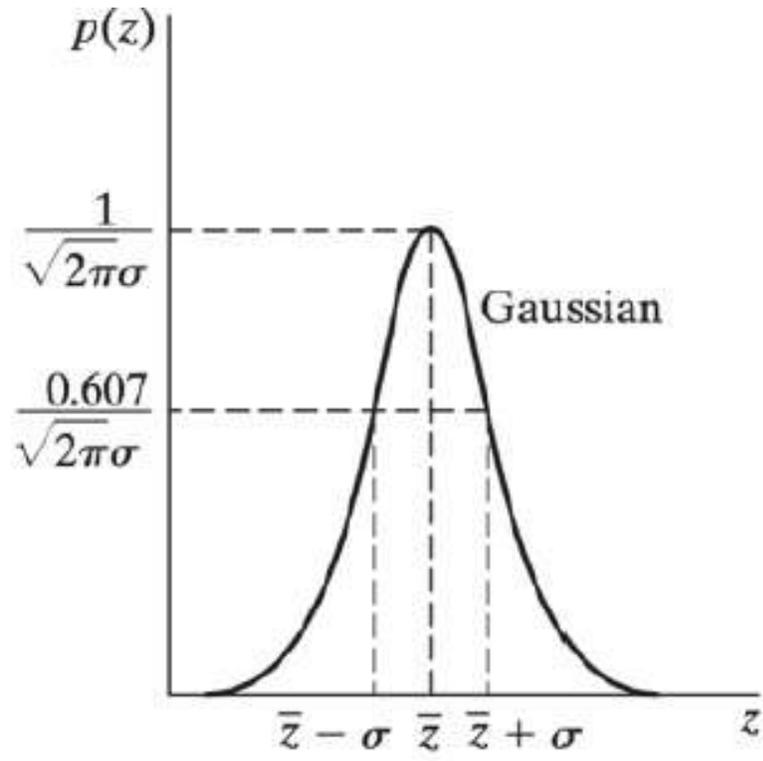


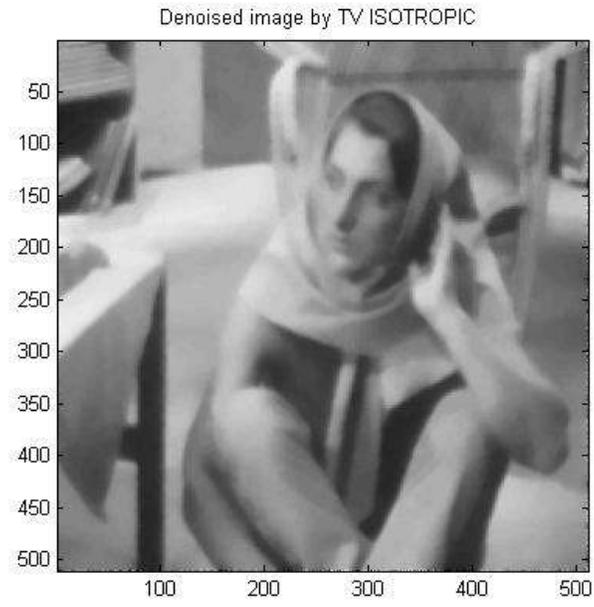
Original image



Noisy image





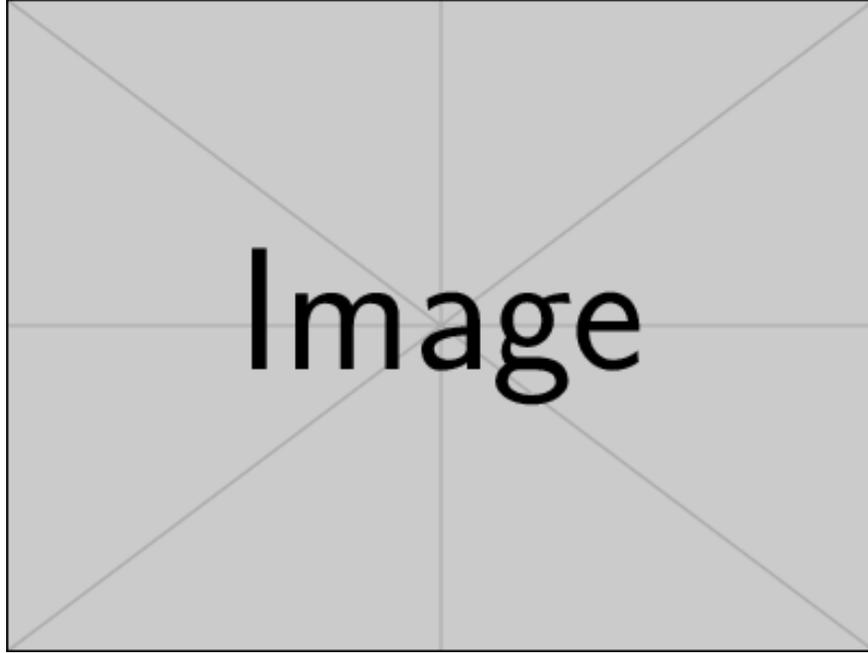


Denoised image by ROF



Denoised image by ROF

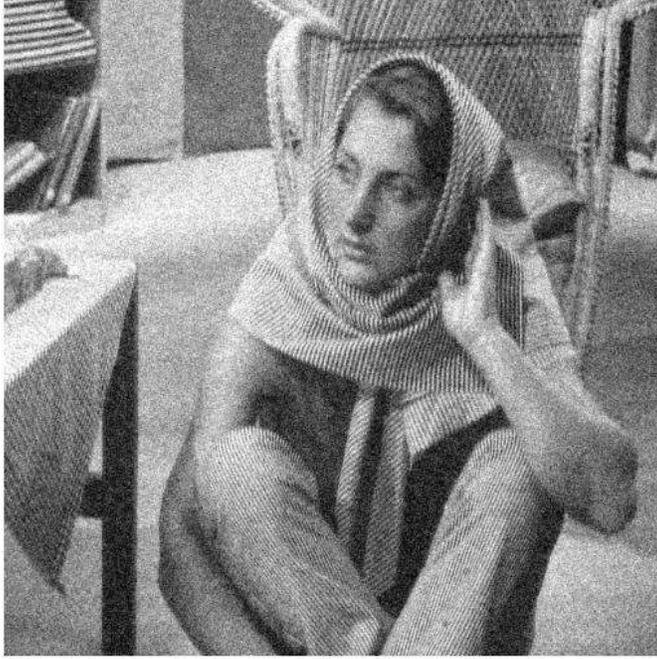




Denoised image by Tyhonov



Denoised image by Tyhonov



ARTICLE TYPE

NUMERICAL COMPARISON OF DIFFERENT IMAGING ALGORITHMS.

Soulef BOUGUEROUA *¹ | Nouredine DAILI^{2,1}

¹Department of Mathematics, Faculty of science, University F. ABBAS Setif 1. 19000 Setif, Algeria, ORCID ID: <https://orcid.org/0000-0001-5724-2009>.

²Department of Mathematics, Faculty of science, University F. ABBAS Setif 1. 19000 Setif, Algeria, ORCID ID: <https://orcid.org/0000-0002-5043-1971>.

Correspondence

*Soulef BOUGUEROUA, Email: soulef.bougueroua@univ-setif.dz

Present Address

University F. ABBAS Setif 1. 19000 Setif, Algeria.

Abstract

Image processing is the set of operations performed to extract "information" from the image. An interesting problem in digital image processing is the restoration of degraded images. It often happens that the resulting image is different from the expected image. Our problem will therefore be to recover an image close to the original image from a poor quality image (that has been skewed by Gaussian and additive noise). There are a lot of algorithms on how we can improve the broken image in better quality. We present in this paper our numerical results obtained with the models of Tichonov regularization, ROF, Vese Osher, anisotropic and isotropic TV denoising algorithms.

KEYWORDS:

Denoising algorithms; Image restoration; Comparative..

1 | INTRODUCTION

A digital image is composed of basic units (referred to as pixels) that each represent a specific area of the image. The width and height of an image are determined by the practically infinite number of pixels that make up each of those dimensions, as well as the range of grayscale or colors that each pixel can contain (we speak of dynamics of the image). There are three categories of digital images:

- **binary images:** in the simplest images, a pixel can only take the values black or white. When a piece of text just has one color, this form of image is typically used to scan it;
- **the grayscale images:** images with gray levels typically display 256 shades of gray. Simply put, each of the 256 colors in a 256 -color image is defined by the range of gray. According to tradition, 0 represents black (null luminous intensity) and 255 represents white (maximum luminous intensity);
- **the color images:** in order to represent the colors red, green, and blue, a color image is actually made up of three images. Each of these three images is referred to as a canal. This representation in red, green, and blue mimics how the human visual system works.

The fundamentally ill-posed character of some practical problems is recognized and is manifested in a very large class of problems, called "inverse problems". There are many types of ill-posed inverse problems, and their applications can be found in many fields such as: Image processing ; Medical imaging; Ultrasound; MRI; Tomography; Geophysical problems; ...

Digital image processing is one of the most crucial components of machine learning or computer vision. A fascinating area of digital image processing is the restoration of images. During the acquisition of a picture (especially through photography), it is typical for the final image to diverge from the expected image. Denoising is the opposite problem from removing noise from an image; the outcome would be subpar if noise was left in the image.

Briefly put, noise is parasitic information that is added to the scene. Since noise has a wide range of origins and characteristics, it can be replicated in many different ways. There are many different kinds of noise, but the case study in this article is Gaussian additive noise with grayscale images, or $f = u + \eta$. The original image is represented by u , the observed noisy image is represented by f , and the Gaussian random fluctuation to zero mean is represented by η . Gaussian noise is often referred to as normal noise in a predefined density function. It is a common technique for including noise in images. According to the following definition, this noise can be produced randomly and separately inside the image with

$$p(z) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}(z-\bar{z})^2}, \quad (1)$$

where z stands for intensity, \bar{z} represents the mean value of z , and σ stands for standard deviation. As seen in **Figure 1**, this function can be visualized. To return the image to a better level of visual quality, descalation techniques are necessary. The investigation of various image restoration models with Gaussian noises will be covered in this paper.

2 | SOME MODELS OF IMAGE RESTORATION

A method for improving or deteriorating an image while reducing its noise content is image restoration. In the sections below, we will go over a few of the image restoration models that will be applied in the experiment.

2.1 | Tichonov Regularization

The oldest regular method still in used to address inverse problems is the Tikhonov regularization method. In other words, we replace the ill-posed original problem with a well-posed alternative approximation problem. It is one of the most well-known methods of regularization in both statistical and digital analysis. The Tichonov regularization is a very commonplace yet overly simplistic regularization method for image processing. If we assume that the additive noise v is Gaussian and that f represents the observed image, then we attempt to reconstruct or restore the image u .

Let $V = H_0^1(\Omega)$ and $H = L^2(\Omega)$, we take the original minimization problem (adjustment to the data):

$$(P) \quad \alpha^* := \min_{u \in V} \|u - f\|_H^2, \quad (2)$$

where $f : \Omega \subset \mathbb{R}^N \rightarrow \mathbb{R}$ is the observed image and the following regularized problem:

$$(P_\alpha) \quad \beta := \min_{u \in V} \{ \|u - f\|_H^2 + \alpha \|\nabla u\|_H^2 \}, \quad \alpha > 0. \quad (3)$$

The gradient must be “very minimal” in order for us to merely adjust u to the data f (it depends on the parameter). A slight gradient, in an image, is “smoothed”. The restoration will provide a blurry image because the margins are eroded.

Theorem 1. Assume that (P) requires at least one answer \tilde{u} . The problem (P_α) requires a one-of-a-kind solution u_α . When $\alpha \rightarrow 0$, one can extract a subsequence from the family $(u_\alpha)_\alpha$ that converges (possibly) in V to a solution u^* of (P) .

The restored image u is far less clear (in particular, the edges are eroded), which makes the problem of image restoration incompatible with the common expression for image restoration, $L(u) = \|\nabla u\|_2^2$ (Tichonov regularization). Think about the overall variation, or consider $L(u) = \int |Du|$. This strategy is significantly more successful. With respect to the problem of functions of bounded variation spaces, this leads to a functional minimization in a particular Banach space.

2.2 | Rudin, Osher, and Fatemi Model

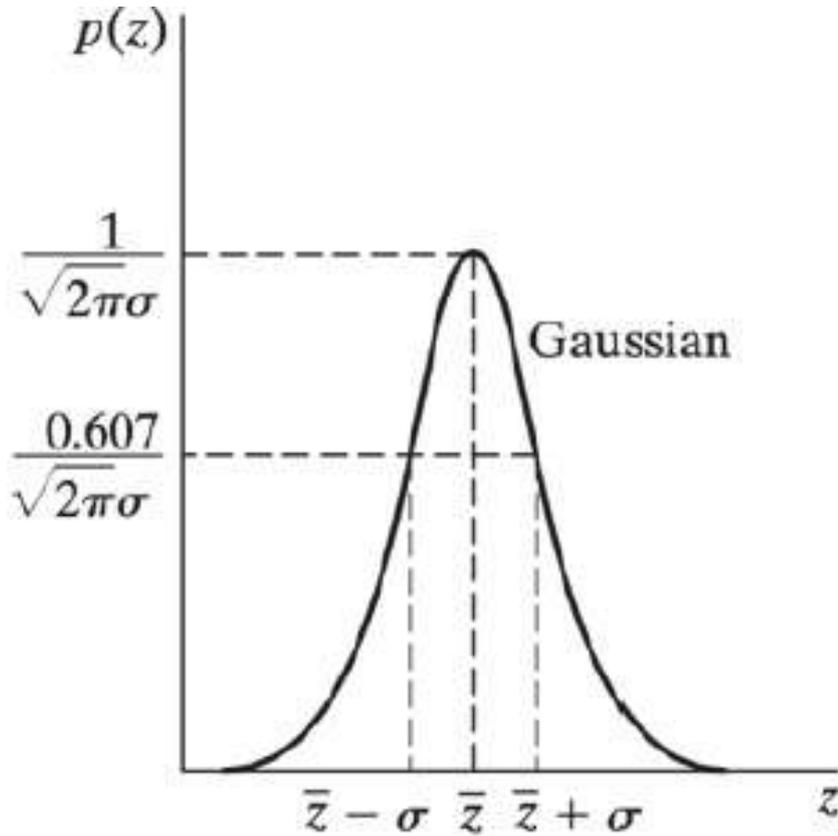


Figure 1 Gaussian Noise.

Rudin, Osher, and Fatemi (*ROF*) proposed the first image restoration model from a given noisy image having additive noise using regularization (*TV*), which is defined by

$$TV(u(x, y)) := \left\{ \int_{\Omega} |\nabla u(x, u)| \, dx dy, \quad \text{with } |\nabla u| = \sqrt{u_x^2 + u_y^2} \right\}.$$

The regularization of total variation (*TV*) approach of image processing is used to reduce noise from digital images. (*TV*) is a technique that was originally developed by *ROF*, it has since been applied to a multitude of other imaging problems.

Rudin, Osher, and Fatemi developed the method known as (TV) to address the problem of visual degradation. Now, it has been used to solve numerous additional image problems.

In ⁽¹⁰⁾, a model has been proposed by Rudin, Osher, and Fatemi and in which the image is divided into two parts $f = u + v$, where u is an unknown image and v is the noise. f is a brilliant measure that is usual at the beginning of a clean image, and is an agreement parameter. We will thus try to solve the problem and simply apply the regularization to the “bruit” portion using the $u + v$ formula with $u \in BV(\Omega)$ and $v \in L^2(\Omega)$. If $f \in L^2$ is correct, the ROF problem is well-posed and the minimizer u exists, is unique, and stable in $L^2(\Omega)$. ROF proposed the following minimization problem:

$$(\mathcal{P}_{ROF}) \quad \alpha_{RoF} := \inf_u \left\{ J(u) + \frac{1}{2\lambda} \|v\|_2^2 : u \in BV(\Omega), v \in L^2(\Omega), f = u + v \right\}. \quad (4)$$

This results in a $[BV(\Omega), L^2(\Omega)]$ decomposition of the image f .

$J(u)$ denotes the total variance of u and $\lambda > 0$.

$$J(u) = \sup \left\{ \int_{\Omega} u(x) \operatorname{div}(\varphi(x)) dx : \varphi \in C_c^1(\Omega, \mathbb{R}^2), \|\varphi\|_{\infty} \leq 1 \right\}.$$

Also known as BV , or functions of bounded variation space, according to

$$BV(\Omega) = \{u \in L^1(\Omega), J(u) < +\infty\}.$$

Here $J(u)$ denotes the TV of u and $\lambda > 0$ is a weight parameter.

Theorem 2. ^(10, 4) The problem (\mathcal{P}_{ROF}) requires a single solution, which is provided by

$$u = f - \lambda \Pi_{\lambda K}(f), \quad (5)$$

where Π is the orthogonal projector on λK (dilatation of K by λ), and K is the overall closure in L^2 .

$$K := \{ \operatorname{div}(\varphi) : \varphi \in C_c^1(\Omega, \mathbb{R}^2), \|\varphi\|_{\infty} \leq 1 \}.$$

2.3 | Meyer's Model

In ⁽⁷⁾, Yves Meyer shows that if λ is small enough, the ROF model will erase the texture. Yves Meyer suggests the use of a space of functions, which is in some ways the dual of the BV space, to extract both the u component in BV and the v component as an oscillating function (texture or noise) from f . The following definition is given by Meyer:

Definition 1. ⁽⁷⁾, $G(\mathbb{R}^2)$ is a Banach space made of v distributions that may be written

$$G(\mathbb{R}^2) = \{v(x, y) = \partial_x g_1(x, y) + \partial_y g_2(x, y) / g_1, g_2 \in L^{\infty}(\mathbb{R}^2)\},$$

We will see that the space G allows for oscillating functions v , as justified by Meyer, and that the oscillations are well measured by the norm

$$\|v\|_G := \inf \left\{ \begin{array}{l} \|g\|_{L^{\infty}(\mathbb{R}^2)} = \operatorname{ess\,sup}_{x \in (\mathbb{R}^2)} |g(x)| / v = \operatorname{div} \vec{g}, \vec{g} = (g_1, g_2) \in L^{\infty}(\mathbb{R}^2) \times L^{\infty}(\mathbb{R}^2), \\ |\vec{g}| = \sqrt{g_1^2 + g_2^2} \end{array} \right\}.$$

Meyer, suggests the following new image restoration model:

$$(\mathcal{P}_{Meyer}) \quad \alpha_{Meyer} := \inf_u \{ J(u) + \alpha \|v\|_G / u \in BV(\Omega), v \in G(\Omega); f = u + v \}. \quad (6)$$

$J(u) = \int |\nabla u|$ denotes the total variation of u and $\alpha > 0$, while $G(\mathbb{R}^2)$ denotes the space of oscillating functions.

2.3.1 | Description of the Model

The interest in this space of oscillating functions stems from the fact that a strongly oscillating image with a small average norm in $G(\mathbb{R}^2)$, can have large oscillations but a small average norm, and that the $L^2(\Omega)$ norm is not the best choice for capturing

the oscillating portion of an image. That is why he created a new space that was better suited from the start: the G oscillating functions space. We have $\|\cdot\|_G$ low for oscillating functions with a null average and high for geometric functions.

We have arrived at the following conclusion based on a close approximation of the L^∞ standard:

$$\left\| \sqrt{g_1^2 + g_2^2} \right\|_{L^\infty} = \lim_{p \rightarrow \infty} \left\| \sqrt{g_1^2 + g_2^2} \right\|_{L^p}, \quad g_1, g_2 \in L^\infty(\mathbb{R}^2).$$

Then, if $\lambda, \mu > 0$ are tuning parameters, $\lambda \rightarrow \infty$ and $p \rightarrow \infty$ the approximation of Meyer model is given by

$$\inf_{u, g_1, g_2} \left\{ G_p(u, g_1, g_2) = \int |\nabla u| + \lambda \int |f - u - \partial_x g_1 - \partial_y g_2|^2 dx dy + \mu \left[\left(\sqrt{g_1^2 + g_2^2} \right)^p \right]^{\frac{1}{p}} \right\}. \quad (7)$$

where

$$\int |\nabla u| \text{ insures that } u \in BV(\mathbb{R}^2),$$

$$\int |f - u - \partial_x g_1 - \partial_y g_2|^2 dx dy \text{ insures that } f \approx u + \text{div}(\vec{g}),$$

$$\mu \left[\left(\sqrt{g_1^2 + g_2^2} \right)^p \right]^{\frac{1}{p}} \text{ is a penalty on the norm } v = \text{div}(\vec{g}) \text{ in } G.$$

As a result, the form of the Euler-Lagrange equation is given here.

$$u = f - \partial_x g_1 - \partial_y g_2 + \frac{1}{2\lambda} \text{div} \left(\frac{\nabla u}{|\nabla u|} \right). \quad (8)$$

$$\mu \left(\left\| \sqrt{g_1^2 + g_2^2} \right\|_p \right)^{1-p} \left(\sqrt{g_1^2 + g_2^2} \right)^{p-2} g_1 = 2\lambda \left[\frac{\partial}{\partial x}(u - f) + \partial_{xx}^2 g_1 + \partial_{xy}^2 g_2 \right]. \quad (9)$$

$$\mu \left(\left\| \sqrt{g_1^2 + g_2^2} \right\|_p \right)^{1-p} \left(\sqrt{g_1^2 + g_2^2} \right)^{p-2} g_2 = 2\lambda \left[\frac{\partial}{\partial y}(u - f) + \partial_{xy}^2 g_1 + \partial_{yy}^2 g_2 \right]. \quad (10)$$

2.4 | Vese-Osher Model

Vese and Osher, who were the first to propose an approach to solve Meyer's problem numerically; that is to say, to realize the program, they used the approximation of Meyer model as follows:

$$(\mathcal{P}_{Vese_Osher}) \quad \alpha_{Vese_Osher} := \inf_{(u, v) \in BV \times G(\Omega)} \left\{ \begin{array}{l} J(u) + \lambda \|f - u - v\|_2^2 + \mu \|v\|_G \\ / u \in BV(\Omega), v \in G(\Omega); u + v = f \end{array} \right\}. \quad (11)$$

In our numerical calculations, the steps to calculate the solution of this problem are:

- 1 - replace the term $\|v\|_G$ by $\left\| \sqrt{g_1^2 + g_2^2} \right\|_p$ with $v = \text{div}(g_1, g_2)$;
- 2 - $p = 1$ is used for digital resonances because it allows for faster calculations each iteration;
- 3 - gives the equation of Euler-Lagrange;
- 4 - we apply a fixed point iterative technique with a finite difference semi-implicit scheme.

2.4.1 | The Numerical Discretization of Meyer's Model

The numerical discretization of equations (8), (9) and (10) is performed using the semi-implicit method of difference and the iterative algorithm based on the fixed point. We used the following initial values for the iterative algorithm:

$$h = 1, \quad p = 1 \text{ and } n = 100.$$

$$\begin{cases} u_0 = f; \\ g_1^0 = -\frac{1}{2\lambda} \frac{f_x}{|\nabla f|}; \\ g_2^0 = -\frac{1}{2\lambda} \frac{f_y}{|\nabla f|}. \end{cases}$$

The following concepts are used: $u_{i,j} = u(ih, jh)$, $f_{i,j} = f(ih, jh)$, $g_{1,i,j} = g_1(ih, jh)$, with the step $h > 0$ and the point (ih, jh) for all $0 \leq i, j \leq M$, and the variable change is taken.

$$H(g_1, g_2) = \left(\left\| \sqrt{g_1^2 + g_2^2} \right\|_p \right)^{1-p} \left(\sqrt{g_1^2 + g_2^2} \right)^{p-2}.$$

So the discretization of equations (8),(9) and (10) is given by

$$u_{i,j}^{n+1} = \frac{1}{1 + \frac{1}{2\lambda h^2}(c_1 + c_2 + c_3 + c_4)} \left(\begin{aligned} & f_{i,j} - \frac{g_{1,i+1,j}^n - g_{1,i-1,j}^n}{2h} - \frac{g_{2,i,j+1}^n - g_{2,i,j-1}^n}{2h} \\ & + \frac{1}{2\lambda h^2}(c_1 u_{i+1,j}^n + c_2 u_{i-1,j}^n + c_3 u_{i,j+1}^n + c_4 u_{i,j-1}^n) \end{aligned} \right); \quad (12)$$

$$g_{1,i,j}^{n+1} = \frac{2\lambda}{\mu H(g_{1,i,j}, g_{2,i,j}) g_{1,i,j}} \left(\begin{aligned} & \frac{u_{i+1,j}^n - u_{i-1,j}^n}{2h} - \frac{f_{i+1,j} - f_{i-1,j}}{2h} + \frac{g_{1,i+1,j}^{n+1} - 2g_{1,i,j}^{n+1} + g_{1,i-1,j}^n}{h^2} \\ & + \frac{1}{4h^2}(g_{2,i+1,j+1}^n + g_{2,i-1,j-1}^n - g_{2,i+1,j-1}^n - g_{2,i-1,j+1}^n) \end{aligned} \right); \quad (13)$$

$$g_{2,i,j}^{n+1} = \frac{2\lambda}{\mu H(g_{1,i,j}, g_{2,i,j}) g_{2,i,j}} \left(\begin{aligned} & \frac{u_{i,j+1}^n - u_{i,j-1}^n}{2h} - \frac{f_{i,j+1} - f_{i,j-1}}{2h} + \frac{g_{2,i,j+1}^{n+1} - 2g_{2,i,j}^{n+1} + g_{2,i,j-1}^n}{h^2} \\ & + \frac{1}{4h^2}(g_{1,i+1,j+1}^n + g_{1,i-1,j-1}^n - g_{1,i+1,j-1}^n - g_{1,i-1,j+1}^n) \end{aligned} \right). \quad (14)$$

The following notations are used:

$$\begin{aligned} c_1 &= \frac{1}{\sqrt{\left(\frac{u_{i+1,j}^n - u_{i,j}^n}{h}\right)^2 + \left(\frac{u_{i,j+1}^n - u_{i,j-1}^n}{2h}\right)^2}}; \\ c_2 &= \frac{1}{\sqrt{\left(\frac{u_{i,j}^n - u_{i-1,j}^n}{h}\right)^2 + \left(\frac{u_{i-1,j+1}^n - u_{i-1,j-1}^n}{2h}\right)^2}}; \\ c_3 &= \frac{1}{\sqrt{\left(\frac{u_{i+1,j}^n - u_{i-1,j}^n}{2h}\right)^2 + \left(\frac{u_{i,j+1}^n - u_{i,j}^n}{h}\right)^2}}; \\ c_4 &= \frac{1}{\sqrt{\left(\frac{u_{i+1,j-1}^n - u_{i-1,j-1}^n}{2h}\right)^2 + \left(\frac{u_{i,j}^n - u_{i,j-1}^n}{h}\right)^2}}. \end{aligned} \quad (15)$$

2.4.2 | Solution of Vese-Osher Problem

In order to solve the Vese-Osher problem, we will study this final problem in the discriminating case, when the image is a vector with two dimensions of size $N \times N$, the Eulidian space $X = \mathbb{R}^{N \times N}$ and $Y = X \times X$.

If $u \in X$ then $\nabla u \in Y$ is defined by $(\nabla u)_{i,j} = \left((\nabla u)_{i,j}^1, (\nabla u)_{i,j}^2 \right)$, where

$$(\nabla u)_{i,j}^1 = \begin{cases} u_{i+1,j} - u_{i,j} & \text{if } i < N, \\ 0 & \text{if } i = N, \end{cases} \quad \text{and } (\nabla u)_{i,j}^2 = \begin{cases} u_{i,j+1} - u_{i,j} & \text{if } j < N, \\ 0 & \text{if } j = N. \end{cases} \quad (16)$$

In the discriminating case, the total variance (TV) of u is defined as

$$J_d(u) = \sum_{1 \leq i, j \leq N} |(\nabla u)_{i,j}|.$$

The divergence operator is $div = -\nabla^*$ (the adjoint of ∇) where

$$(div(p))_{i,j} = \begin{cases} p_{i,j}^1 - p_{i-1,j}^1 & \text{if } 1 < i < N \\ p_{i,j}^1 & \text{if } i = 1 \\ -p_{i-1,j}^1 & \text{if } i = N \end{cases} + \begin{cases} p_{i,j}^2 - p_{i,j-1}^2 & \text{if } 1 < j < N \\ p_{i,j}^2 & \text{if } j = 1 \\ -p_{i,j-1}^2 & \text{if } j = N, \end{cases} \quad (17)$$

and the G^d -space is defined by

$$G^d := \{v \in X / \exists g \in Y \text{ such that } v = div(g)\}.$$

Note that

$$G_\mu(\Omega) := \{v \in G(\Omega) \text{ such that } \|v\|_G \leq \mu\},$$

$$G_\mu^d(\Omega) := \{v \in G^d(\Omega) \text{ such that } \|v\|_{G^d} \leq \mu\}.$$

As J_d^* is the indicator function of $G_1^d(\Omega)$ defined by

$$J_d^*(v) = \chi_{G_1^d}(v) = \begin{cases} 0 & \text{if } v \in G_1^d \\ +\infty & \text{else.} \end{cases}$$

So to solve the Vese-Osher problem, we propose the following algorithm

Algorithm 1 The algorithm for solving Vese Osher problem

$$\inf_{(u,v) \in BV(\Omega) * G_\mu(\Omega)} \left\{ F_{\lambda,\mu}(u,v) = \begin{cases} J(u) + \frac{1}{2\lambda} \|f - u - v\|_2^2 & \text{if } v \in G_\mu(\Omega) \\ +\infty & \text{if } v \in G(\Omega)/G_\mu(\Omega) \end{cases} \right\}.$$

By description;

$$\inf_{(u,v) \in X * X} \left\{ F_{\lambda,\mu}(u,v) = \begin{cases} J_d(u) + \frac{1}{2\lambda} \|f - u - v\|_X^2 & \text{if } v \in G_\mu^d(\Omega) \\ +\infty & \text{if } v \in X/G_\mu^d(\Omega) \end{cases} \right\}.$$

$$\inf_{(u,v) \in X * X} F(u,v) = \inf_{(u,v) \in X * X} \left\{ J_d(u) + \frac{1}{2\lambda} \|f - u - v\|_X^2 + J_d^*\left(\frac{v}{\mu}\right) \right\}.$$

We divide the problem into two sub-problems:

$$Pbm1 \quad u \text{ solution, } v \text{ fixed: } \inf_{v \in X} \left\{ J_d(u) + \frac{1}{2\lambda} \|f - u - v\|_X^2 \right\}.$$

$$\curvearrowright_{\substack{\text{the solution is} \\ \text{according to ROF} \\ \text{change of variable} \\ f=f-v}} u^l = f - v - P_{G_\mu^d}(f - v).$$

$$Pbm2 \quad v \text{ solution, } u \text{ fixed: } \inf_{v \in G_\mu^d(\Omega)} \left\{ \|f - u - v\|_X^2 \right\}.$$

$$\curvearrowright_{\substack{\text{the solution is} \\ \text{according to ROF} \\ \text{change of variable} \\ f=f-u}} v^l = P_{G_\mu^d}(f - u).$$

Lemma 1. There is an unique solution $(\hat{u}, \hat{v}) \in X * G_\lambda^d$ that minimizes $F_{\lambda,\mu}(u, v)$ in $X * G_\lambda^d$.

2.5 | The Split Bregman Algorithm

Goldstein and Osher first proposed the split Bregman algorithm in ⁽¹²⁾ to handle more general form optimization problems:

$$\varpi := \min_{u \in X} \{H(u) + \|\Phi(u)\|_1\}, \quad (18)$$

where X is a closed convex set and $\Phi : X \rightarrow \mathbb{R}$, $H : X \rightarrow \mathbb{R}$ are convex functions. This problem is the same as the stress minimization problem as below:

$$\overline{\varpi} := \min_{u \in X, d \in \mathbb{R}} \{H(u) + \|d\|_1\} \quad \text{such that } d = \Phi(u). \quad (19)$$

Goldstein and Osher introduced the split Bregman algorithm, which was written as follows: The split Bregman algorithm is

Algorithm 2 The split Bregman algorithm

Initialization: $k = 0, u^0 = 0, b^0 = 0$
 While $\|u^k - u^{k-1}\| > tol$ do
 $u^{k+1} = \min_u H(u) + \frac{\lambda}{2} \|d^k - \Phi(u) - b^k\|_2^2$
 $d^{k+1} = \min_d |d| + \frac{\lambda}{2} \|d - \Phi(u^{k+1}) - b^k\|_2^2$
 $b^{k+1} = b^k + (\Phi(u^{k+1}) - d^{k+1})$
 $k = k + 1$
 End while.

used to solve some of the most common form optimization problems:

$$\hat{w} := \min_{u \in X} \left\{ z(u) + \frac{1}{2} \|u - f\|_2^2 \right\}. \quad (20)$$

Anisotropic and isotropic TV denoising problems are solved using the split Bregman method.

2.5.1 | Anisotropic TV Denoising Problem

The problem of anisotropic TV denoising is considered in ⁽³⁾.

$$(\mathcal{P}_1) \quad \tau_1 := \min_u \left\{ \left\| \frac{\partial u}{\partial x} \right\|_1 + \left\| \frac{\partial u}{\partial y} \right\|_1 + \frac{\mu}{2} \|u - f\|_2^2 \right\}, \quad (21)$$

where f is the noisy image, $\frac{\partial u}{\partial x}$ and $\frac{\partial u}{\partial y}$ will be noted by u_x and u_y respectively. The problem is solved using a constraint equivalent to a problem (\mathcal{P}_1) .

We answer the problem (\mathcal{P}_2) as follows:

$$(\mathcal{P}_2) \quad \begin{cases} \tau_2 := \min_u \|d_x\|_1 + \|d_y\|_1 + \frac{\mu}{2} \|u - f\|_2^2 \\ \text{subject to } d_x = u_x, d_y = u_y. \end{cases}$$

The Split Bregman algorithm can be used to tackle this last problem:

$$(\mathcal{P}_3) \quad \tau_3 := \min_{u, d_x, d_y} \left\{ \|d_x\|_1 + \|d_y\|_1 + \frac{\mu}{2} \|u - f\|_2^2 + \frac{\lambda}{2} \|d_x - u_x\|_2^2 + \frac{\lambda}{2} \|d_y - u_y\|_2^2 \right\}. \quad (22)$$

We use

$$\text{shrink}(x, a) = \begin{cases} x - a & \text{if } x > a, \\ x + a & \text{if } x < -a, \\ 0 & \text{else.} \end{cases} \quad (23)$$

The Gauss-Seidel function is also useful.

$$G_{i,j}^k = \frac{\lambda}{\mu + 4\lambda} (u_{i+1,j}^k + u_{i-1,j}^k + u_{i,j+1}^k + u_{i,j-1}^k + d_{x,i-1,j}^k + d_{x,i,j}^k + d_{y,i,j-1}^k + d_{y,i,j}^k + b_{x,i-1,j}^k + b_{x,i,j}^k + b_{y,i,j-1}^k + b_{y,i,j}^k) + \frac{\mu}{\mu + 4\lambda} f_{i,j}. \quad (24)$$

2.5.2 | Isotropic TV Denoising Problem

The problem of isotropic TV denoising is considered in ⁽³⁾,

$$(\mathcal{P}'_1) \quad I_{S_1} := \min_u \left\{ \|\nabla u\|_2 + \frac{\mu}{2} \|u - f\|_2^2 \right\}. \quad (25)$$

Algorithm 3 The split Bregman algorithm of anisotropic TV denoising

Initialization: $k = 0, u^0 = 0, b^0 = 0$.
While $\|u^k - u^{k-1}\| > tol$ do,
 $u^{k+1} = G^k$, where G is the Gauss-Seidel function.
 $d_x^{k+1} = shrink(\nabla_x u^{k+1} + b_x^k, \frac{1}{\lambda})$.
 $d_y^{k+1} = shrink(\nabla_y u^{k+1} + b_y^k, \frac{1}{\lambda})$.
 $b_x^{k+1} = b_x^k(\nabla_x u^{k+1} - d_x^{k+1})$.
 $b_y^{k+1} = b_y^k(\nabla_y u^{k+1} - d_y^{k+1})$.
 $k = k + 1$.
End while.

The problem (P'_1) is solved using a constraint equivalent problem (P'_2) :

$$(P'_2) \quad \begin{cases} \widetilde{I}S_1 := \min_u \|(d_x, d_y)\|_2 + \frac{\mu}{2} \|u - f\|_2^2 \\ \text{subject to } d_x = u_x, d_y = u_y. \end{cases} \quad (26)$$

To solve the problem (P'_2) , we solve the following problem without constraint:

$$(P'_3) \quad \widetilde{I}S_3 := \min_{u, d_x, d_y} \left\{ \|(d_x, d_y)\|_2 + \frac{\mu}{2} \|u - f\|_2^2 + \frac{\lambda}{2} \|d_x - u_x\|_2^2 + \frac{\lambda}{2} \|d_y - u_y\|_2^2 \right\}. \quad (27)$$

The Split Bregman algorithm can be used to tackle this last difficulty.

We give the following definitions:

$$s^k = \sqrt{|u_x^k - b_x^k|^2 + |u_y^k - b_y^k|^2}. \quad (28)$$

Algorithm 4 The split Bregman algorithm of isotropic TV denoising

Initialization: $k = 0, u^0 = 0, b^0 = 0$.
While $\|u^k - u^{k+1}\| > tol$ do,
 $u^{k+1} = G^k$, where G is the Gauss-Seidel function.
 $d_x^{k+1} = \frac{s^k \lambda (u_x^k + b_x^k)}{s^k \lambda + 1}$.
 $d_y^{k+1} = \frac{s^k \lambda (u_y^k + b_y^k)}{s^k \lambda + 1}$.
 $b_x^{k+1} = b_x^k + (u_x^{k+1} - d_x^{k+1})$.
 $b_y^{k+1} = b_y^k + (u_y^{k+1} - d_y^{k+1})$.
 $k = k + 1$.
End while.

3 | NUMERICAL EXPERIMENTAL RESULTS

We present in this section our numerical results obtained with the following models of: Tichonov regularization, *ROF*, anisotropic and isotropic *TV* denoising. Let X be the matrices that depict an image of size $m \times n$. We then used Matlab $f = imnoise(X, 'gaussian', sigma)$ command to define our noise image f , where sigma is a version of the Gaussian noise level. We used the values $\mu = 0.1$, $\lambda = 0.2$ and the tolerance $Tol = 10^{-5}$ in our studies. In our experience, we have tried to implement several models of rehabilitation. Each model aims to produce a better solution to remove noise from the image. However, we are going to implement iast in the script. By calculating Performance metrics as well different sigma values, we try to present

Table 1 Results for the ROF algorithm, $\sigma = 0.01$.

Images noisyPSNR	Size $n \times m$	Number of iterations	$\frac{\ denoised I - noisy I\ ^2}{(MN * \sigma^2) - 1}$	denoised PSNR
cameraman 20.3181 †	398×398	25	0.1349	24.6132
barbara 20.1075	510×510	25	0.41724	22.3979
camera 20.4143	340×340	26	0.1179	24.9959
flower 20.1339	256×256	22	0.099939	25.9711
girl 20.2102	216×233	21	0.018344	26.1987
Iline 19.9895	1961×3553	24	34.5489	34.5489
university 20.554 ‡	480×640	25	0.050125	25.0007

the best result. The results for Tichonov regularization, ROF, anisotropic and isotropic TV denoising algorithms are in Tables 1; 2, ; 3; 4; 5; 6; 7 and 8.

In addition, in Tables 9; 10; 11; 12; 13; 14; 15 and 16, we evaluate quality of images restored by the image restoration models, we use square error (MSE), signal noise rate (SNR), peak signal to noise ratio ($PSNR$), image quality index (IQI), normalized cross-correlation (NK), average difference (AD), structural content (SC), maximum difference (MD), and normalized absolute error (NAE); of Tichonov regularization, ROF, anisotropic and isotropic TV denoising algorithms.

In Figure 2, we did an experiment by taking the original image of Barbara (image without noise), then we added white Gaussian noise (sigma 0.08).

A comparative numerical was carried out between the Tichonov regularization restoration model and the ROF; TV anisotropic and isotropic denoising algorithm for the same parameter sigma 0.08 is shown in Figures 3, 4.

In Figure 5, we did an experiment by taking the original image of girl (image without noise), then we added white Gaussian noise (sigma 0.08).

Finally, in Figures 7 and 8; we show comparisons and numerical results for the Tichonov regularization restoration model and the ROF model; TV anisotropic and isotropic denoising algorithm with a noisy image of girl for the same parameter sigma 0.08.

Table 2 Results for the ROF algorithm, $\sigma = 0.08$.

Images noisyPSNR	Size $n \times m$	Number of iterations	$\frac{\ denoised I - noisy I\ ^2}{(MN * \sigma^2) - 1}$	denoised PSNR
cameraman 20.3181 †	398×398	25	0.1349	24.6132
barbara 20.1075	510×510	25	0.41724	22.3979
camera 18.8938	340×340	29	-0.87675	20.7695
flower 20.1339	256×256	22	0.099939	25.9711
girl 18.1269	216×233	20	-0.87384	20.7782
Iline 17.9203	1961×3553	29	-0.87735	21.8224
university 18.5907 ‡	480×640	26	-0.8717	20.6516

Table 3 Results for the ROF algorithm, $\sigma = 0.2$.

Images noisyPSNR	Size $n \times m$	Number of iterations	$\frac{\ denoised I - noisy I\ ^2}{(MN * \sigma^2) - 1}$	denoised PSNR
cameraman 13.2205 †	398×398	38	-0.94385	13.7602
barbara 13.2581	510×510	30	-0.93372	13.5776
camera 15.0525	340×340	38	-0.9668	15.234
flower 13.078	256×256	34	-0.94424	13.7758
girl 13.4558	216×233	25	-0.95348	14.0803
Iline 13.2012	1961×3553	32	-0.95397	14.0965
university 13.8881	480×640	35	-0.95166	14.3564
university 20.554 ‡	480×640	25	0.050125	25.0007

Table 4 Results for the Tichonov Regularization algorithm, sigma = 0.01

Images	Size $n \times m$	$\frac{\ denoised\ I-noisy\ I\ ^2}{(MN*sigma^2)-1}$	denoised PSNR	noisyPSNR
cameraman	398 × 398	-0.97352	23.9646	20.3181 †
barbara	510 × 510	-0.96845	23.6274	20.1075
camera	340 × 340	-0.97294	24.0817	20.4143
flower	256 × 256	-0.97523	24.1745	20.1339
girl	216 × 233	-0.97692	24.3646	20.2102
Iline	1961 × 3553	-0.97614	24.2821	19.9895
university	480 × 640	-0.97551	24.3021	20.554
‡				

Table 5 Results for the Tichonov Regularization algorithm, sigma = 0.08

Images	Size $n \times m$	$\frac{\ denoised\ I-noisy\ I\ ^2}{(MN*sigma^2)-1}$	denoised PSNR	noisyPSNR
cameraman	398 × 398	-0.97352	23.9646	20.3181 †
barbara	510 × 510	-0.96845	23.6274	20.1075
camera	340 × 340	-0.97294	24.0817	20.4143
flower	256 × 256	-0.97523	24.1745	20.1339
girl	216 × 233	-0.97692	24.3646	20.2102
Iline	1961 × 3553	-0.97614	24.2821	19.9895
university	480 × 640	-0.97551	24.3021	20.554
‡				

Table 6 Results for the Tichonov Regularization algorithm, sigma = 0.08

Images	Size $n \times m$	$\frac{\ denoised\ I-noisy\ I\ ^2}{(MN*sigma^2)-1}$	denoised PSNR	noisyPSNR
cameraman	398 × 398	-0.97254	19.8452	17.9764 †
barbara	510 × 510	-0.96842	19.7739	17.9621
camera	340 × 340	-0.97612	20.7023	18.8938
flower	256 × 256	-0.97468	19.9527	17.9206
girl	216 × 233	-0.9771	20.2027	18.1269
Iline	1961 × 3553	-0.97636	20.0426	17.9203
university	480 × 640	-0.97617	20.5036	18.5907
‡				

Table 7 Results for the Tichonov Regularization algorithm, sigma = 0.2

Images	Size $n \times m$	$\frac{\ denoised I - noisy I\ ^2}{(MN * sigma^2) - 1}$	denoised PSNR	noisyPSNR
cameraman	398×398	-0.97377	13.7374	13.2205 †
barbara	510×510	-0.97051	13.7499	13.2581
camera	340×340	-0.98386	15.4174	15.0525
flower	256×256	-0.9748	13.6479	13.078
girl	216×233	-0.97887	14.0148	13.4558
Iline	1961×3553	-0.97781	13.769	13.2012
university	480×640	-0.97776	14.4049	13.8881 ‡

Table 8 Results for the anisotropic TV denoising algorithm, sigma = 0.08.

Images	Size $n \times m$	Number of iterations	Relative Error	Time (s)
cameraman	398×398	43	0.104779	45.680272 †
barbara	510×510	41	0.173054	85.609672
camera	340×340	7	0.165105	5.001468
flower	256×256	51	0.211968	24.340052
girl	216×233	47	0.149544	17.045064
Iline	1961×3553	164	0.143347	10438.736396
university	480×640	141	0.143296	300.869216 ‡

Table 9 Results for the isotropic TV denoising algorithm, sigma = 0.08.

Images	Size $n \times m$	Number of iterations	Relative Error	Time (s)
cameraman	398×398	23	0.104677	57.237110 †
barbara	510×510	137	0.17464	182.507988
camera	340×340	143	0.172107	86.802806
flower	256×256	22	0.210941	41.570893
girl	216×233	24	0.151122	18.938243
Iline	1961×3553	15	0.143349	771.958677
university	480×640	24	0.143312	120.298105 ‡

4 | CONCLUSION

In this paper, we have presented and compared theoretical and numerical of different imaging algorithms for solving optimization problems. We are looking for an image that is near to the original as possible among images that have been skewed by Gaussian and additive noise. Image deconstruction is a technique for restoring a noisy image after it has been captured. According to our experimentation, and by calculating performance metrics as well different sigma values, we can conclude that the ROF model is better image quality compared to the Tichonov regularization, because the space BV ensures continuity and allows for a stairway effect in restoring smooth images in applications where edges are not the main feature. we can conclude that the anisotropic TV and isotropic TV denoising algorithms work in a direct correlation relationship. In other words, regardless

Table 10 Performance metrics for ROF algorithm, $\sigma = 0.01$.

Images NK	MSE AD	SNR SC	PSNR MD	IQI NAE
cameraman	95.4334	3.9647	28.3338	-2.4160e-06
1.3366	-64.3582	0.4536	223	0.6382 †
barbara	96.1126	4.2188	28.3030	4.8606e-08
1.2066	-51.1091	0.4790	223	0.7199
camera	94.6071	4.3055	28.3716	3.1438e-05
1.1302	-37.7750	0.7046	234	0.3002
flower	95.6821	4.2095	28.3225	-7.1457e-07
1.5430	-88.4721	0.2649	182	1.1511
girl	95.0412	4.1605	28.3517	-1.5265e-07
1.3229	-75.2978	0.4226	196	0.7586
Iline	95.9190	4.2459	28.3118	0.9893
1.3479	-58.0632	0.4805	206	0.5112
university	88.4816	4.5937	28.6623	7.7624e-04
1.0743	-46.5255	0.5581	250	0.6777 ‡

Table 11 Performance metrics for ROF algorithm, $\sigma = 0.08$.

Images NK	MSE AD	SNR SC	PSNR MD	IQI NAE
cameraman	95.4334	3.9647	28.3338	-2.4160e-06
1.3366	-64.3582	0.4536	223	0.6382 †
barbara	96.1126	4.2188	28.3030	4.8606e-08
1.2895	-62.4257	0.4411	223	0.7338
camera	162.7122	1.9505	26.0166	3.1438e-05
1.1711	-47.1331	0.6584	234	0.3313
flower	95.6821	4.2095	28.3225	-7.1457e-07
1.5430	-88.4721	0.2649	182	1.1511
girl	163.1718	1.8132	26.0044	-1.5265e-07
1.3696	-82.4141	0.4053	196	0.7715
Iline	164.7801	1.8959	25.9618	0.9893
1.4095	-66.9820	0.4521	188	0.5319
university	151.8053	2.2494	26.3179	7.7624e-04
1.1407	-57.3012	0.5196	250	0.6852 ‡

of how little the sigma value is that we get better and more old image quality results. Finally, it should be mentioned that all the methods that are common for removing parasitic information are added from an image.

Table 12 Performance metrics for ROF algorithm, $\sigma = 0.2$.

Images NK	MSE AD	SNR SC	PSNR MD	IQI NAE
cameraman	238.3766	-0.0110	24.3582	-1.5908e-07
1.4816	-89.4316	0.3828	190	0.7872 †
barbara	239.9024	0.2463	24.3305	4.8606e-08
1.4276	-81.4507	0.3819	223	0.8033
camera	236.3087	0.3299	24.3960	3.1438e-05
1.2246	-59.9995	0.6018	234	0.3897
flower	239.7857	0.2196	24.3326	-7.1457e-07
1.7601	-110.2763	0.2341	182	1.2374
girl	237.7749	0.1780	24.3691	-1.5265e-07
1.4474	-94.2270	0.3762	196	0.8140
liline	239.5990	0.2701	24.3360	0.9893
1.5124	-81.9270	0.4059	160	0.6046
university	219.8879	0.6402	24.7088	7.7624e-04
1.2430	-74.7657	0.4619	250	0.7667 ‡

Table 13 Performance metrics for Tichonov algorithm, $\sigma = 0.01$.

Images NK	MSE AD	SNR SC	PSNR MD	IQI NAE
cameraman	95.4334	3.9647	28.3338	2.1712e-05
1.3361	-64.2595	0.4506	244	0.6637 †
barbara	96.1126	4.2188	28.3030	6.7731e-09
1.2060	-51.0406	0.4732	223	0.7462
camera	94.6071	4.3055	28.3716	3.0292e-05
1.1294	-37.6595	0.7005	241	0.3220
flower	95.6821	4.2095	28.3225	-3.6032e-06
1.5423	-88.3622	0.2636	201	1.1744
girl	95.0412	28.3517	4.1605	2.0528e-07
1.3220	-75.1643	0.4206	201	0.7728
liline	95.9190	4.2459	28.3118	0.9841
1.3469	-57.9367	0.4791	254	0.5287
university	88.4816	4.5937	28.6623	7.1609e-04
1.0743	-46.4538	0.5522	254	0.7034 ‡

DATA AVAILABILITY

No data were used to support this study.

Table 14 Performance metrics for Tichonov algorithm, sigma = 0.08.

Images NK	MSE AD	SNR SC	PSNR MD	IQI NAE
cameraman	164.1012	1.6105	25.9797	-4.2489e-06
1.3902	-73.4494	0.4244	233	0.6934 †
barbara	165.0845	1.8696	25.9537	4.0697e-08
1.2888	-62.3501	0.4363	223	0.7574
camera	162.7122	1.9505	26.0166	3.0208e-05
1.1702	-46.9840	0.6554	235	0.3418
flower	1.8539	164.5883	25.9668	-3.0678e-06
1.6220	-96.3537	0.2529	197	1.1856
girl	163.1718	1.8132	26.0044	2.8824e-07
1.3688	-82.2831	0.4034	202	0.7837
liline	164.7801	1.8959	25.9618	0.9893
1.4084	-66.8355	0.4511	238	0.5417
university	151.8053	2.2494	26.3179	7.7624e-04
1.1404	-57.1934	0.5148	254	0.7062 ‡

Table 15 Performance metrics for Tichonov algorithm, sigma = 0.2.

Images NK	MSE AD	SNR SC	PSNR MD	IQI NAE
cameraman	238.3766	-0.0110	24.3582	1.1434e-06
1.4810	-89.3073	0.3809	221	0.7922 †
barbara	239.9024	0.2463	24.3305	-4.0468e-08
1.4267	-81.3551	0.3786	223	0.8199
camera	236.3087	0.3299	24.3960	3.0276e-05
1.2233	-59.7684	0.6005	234	0.3932
flower	239.7857	0.2196	24.3326	-2.8803e-06
1.7591	-110.1366	0.2331	182	1.2489
girl	237.7749	0.1780	24.3691	-1.5265e-07
1.4463	-94.0664	0.3748	195	0.8222
liline	239.5990	0.2701	24.3360	0.9882
1.5113	-81.7794	0.4052	208	0.6085
university	219.8879	0.6402	24.7088	7.7368e-04
1.2425	-74.6313	0.4586	254	0.7748 ‡

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

Table 16 Performance metrics for the anisotropic TV denoising algorithm, $\sigma = 0.08$.

Images NK	MSE AD	SNR SC	PSNR MD	IQI NAE
cameraman	7.6326e+03	15.9295	9.3041	2.8414e-05
1.1423	-44.5176	0.6582	255	0.3806 †
barbara	1.4142e+04	11.8896	6.6257	-3.5635e-06
1.3828	-77.6333	0.3766	234	0.8772
camera	1.1758e+04	12.4348	7.4276	-7.3644e-05
1.3689	-73.9475	0.4184	228	0.7373
flower	1.6863e+04	10.0637	5.8614	5.5853e-06
1.5642	-88.7960	0.2694	255	1.1090
girl	1.3374e+04	12.9140	6.8682	8.4450e-08
1.3312	-75.0687	0.4254	221	0.7501
l1ne	232.5184	12.9318	24.4662	0.0052
1.0129	-5.3346	0.9706	255	0.0268
university	1.4088e+04	13.1774	6.6423	2.3093e-04
1.1405	-61.5217	0.4907	255	0.7909 ‡

Table 17 Performance metrics for the isotropic TV denoising algorithm, $\sigma = 0.08$.

Images NK	MSE AD	SNR SC	PSNR MD	IQI NAE
cameraman	7.5933e+03	15.8979	9.3265	3.1550e-05
1.1435	-44.1626	0.6577	255	0.3825 †
barbara	1.3937e+04	11.8882	6.6890	-3.3960e-06
1.3831	-77.6695	0.3784	242	0.8666
camera	1.1626e+04	12.4178	7.4764	-7.3644e-05
1.3804	-76.0561	0.4157	228	0.7384
flower	1.6711e+04	10.1155	5.9007	5.1306e-06
1.5626	-88.6839	0.2706	255	1.0941
girl	1.3120e+04	12.9056	6.9516	4.4023e-07
1.3339	-75.3454	0.4267	221	0.7382
l1ne	210.5850	12.9318	24.8965	0.0052
1.0122	-5.0294	0.9724	255	0.0250
university	1.3858e+04	13.1735	6.7138	2.3140e-04
1.1553	-63.5094	0.4866	255	0.7864 ‡

ACKNOWLEDGMENTS

We would like to thank the referees for some corrections which greatly improved the presentation of this paper.

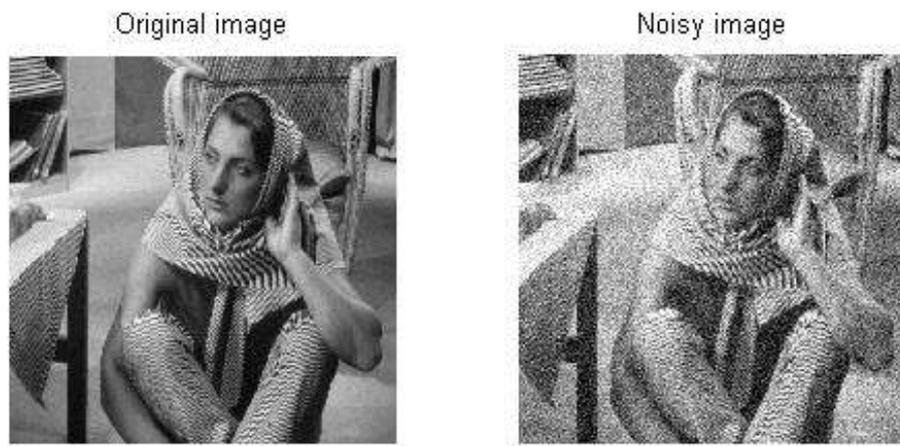


Figure 2 The original and noisy image barbara for $\sigma = 0.08$.



Figure 3 Denoised image barbar by Tikhonov and ROF for sigma = 0.08.

References

1. J.-F. Aujol, G. Aubert, L. Blanc-Féraud and A. Chambolle, Image decomposition into a bounded variation component and an oscillating component, *J. Math. Image Vision* 22 (1) (2005), pp. 71-88.
2. Maïtine Bergounioux, Quelques méthodes mathématiques pour le traitement d'image, DEA, Cours M2, Université d'Orléans 2008-2009, 2008, pp.110. cel-00125868v4
3. Jacqueline Bush, Bregman Algorithms, Senior Thesis, University of California, Santa Barbara, June 10, 2011.
4. A. Chambolle, An Algorithm for Total Variation Minimization and Applications, *J. Math. Imaging Vision* 20 (2004), pp. 89-97.
5. P. Getreuer, Rudin-Osher-Fatemi Total Variation Denoising using Split Bregman, in *Image Processing On Line*, on 2012-05-19. ISSN 2105-1232, c 2012 IPOL & the authors CC-BY-NC-SA This article is available online with supplementary materials, software, data sets and online demo at <https://dx.doi.org/10.5201/ipol.2012.g-tvd>
6. S. Kabbadj, Inexact Version of Bregman Proximal Gradient Algorithm, *Hindawi Abstract and Applied Analysis*, Vol. 2020, Article ID 1963980, 11 pages, <https://doi.org/10.1155/2020/1963980>
7. Y. Meyer, Oscillating patterns in image processing and in some nonlinear evolution equations, *The Fifteenth Dean Jacqueline B. Lewis Memorial Lectures*, March 2001.
8. Ch. A. Micchelli, Lixin Shen and Yuesheng Xux, Proximity algorithms for image models: Denoising, *Article in Inverse Problems*, March 2011, DOI: 10.1088/0266-5611/27/4/045009
9. Osber, A. Solé, and L. Vese, Image Decomposition and Restoration Using Total Variation Minimization and the H^{-1} Norm, *Journal of Multiscale Modeling and Simulation*, Vol. 1, no. 3, 2003, pp. 394-370

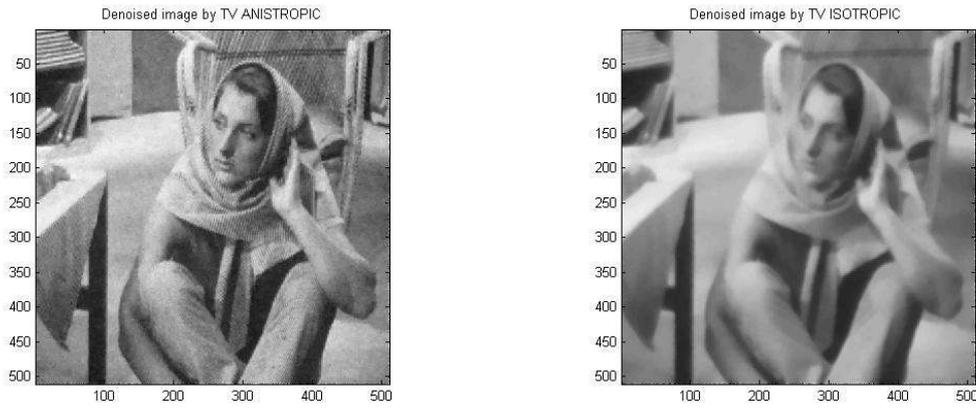


Figure 4 Denoised image barbara by TV anistropic and isotropic for sigma = 0.08.

10. L. Rudin, S. Osher, and E. Fatemi, Total variation based noise removal algorithms, *Physica D*, 60 (1992), pp. 259-268.
11. L. Vese, and S. J. Osher, Textures with total variation minimization and oscillating patterns in image processing, UCLA, C. A. M. Report 02-19, 2002.
12. T. Goldstein and S. Osher, The split Bregman method for ℓ_1 -regularized problems, *SIAM J. Imaging Sci.*, 2(2)(2009), 323-343.



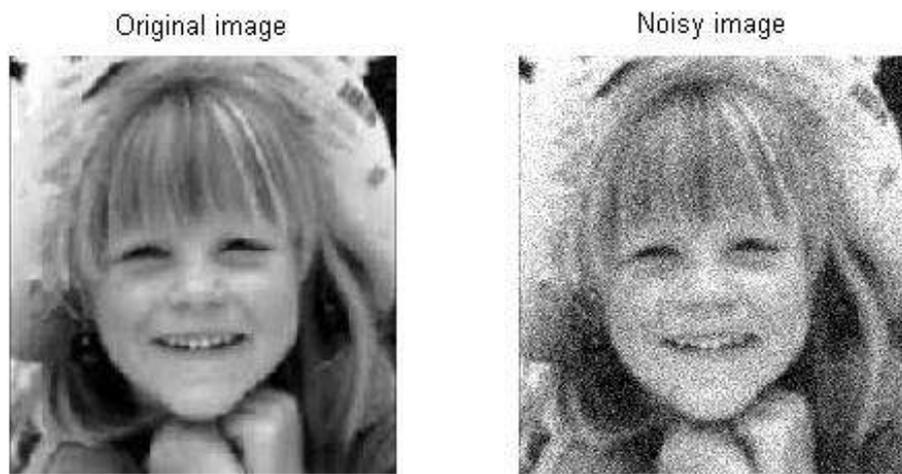


Figure 5 The original and noisy image girl for sigma = 0.08.

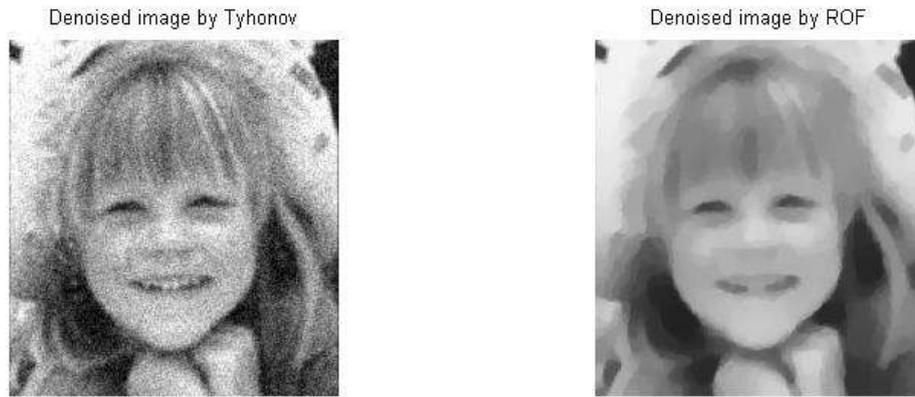


Figure 6 Denoised image girl by Tikhonov and ROF for sigma = 0.08.

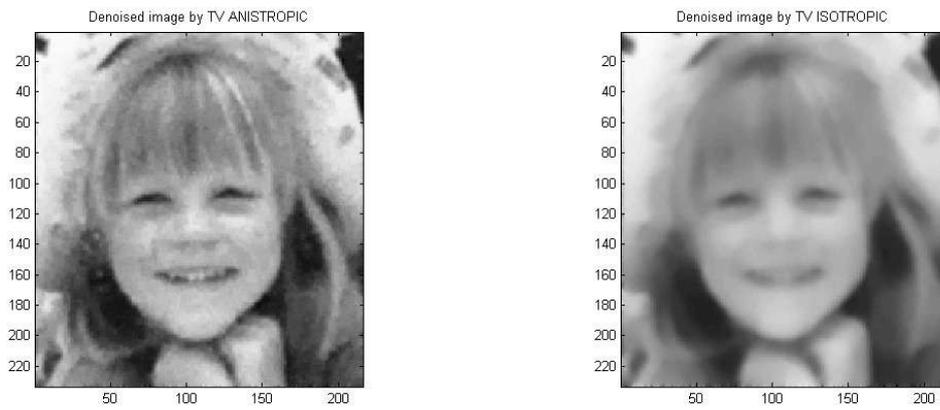


Figure 7 Denoised image girl by TV anistropic and isotropic for sigma = 0.08.