Investigation of Key Parameters to Improve Denoising Images based on Diffusion Equation

This paper was downloaded from TechRxiv (<https://www.techrxiv.org>).

LICENSE

CC BY 4.0

SUBMISSION DATE / POSTED DATE

03-10-2021 / 17-10-2021

CITATION

Torkamani Azar, Farah; Sabaghian, Fatemeh (2021): Investigation of Key Parameters to Improve Denoising Images based on Diffusion Equation. TechRxiv. Preprint. <https://doi.org/10.36227/techrxiv.16727212.v1>

DOI

[10.36227/techrxiv.16727212.v1](https://doi.org/10.36227/techrxiv.16727212.v1)

Investigation of Key Parameters to Improve Denoising Images based on Diffusion Equation

Fatemeh Sabaghian, Farah Torkamani Azar

Abstract—Denoising is as a vital step in image enhancement applications. While Perona and Maliks classic diffusion equation has shown remarkable performance in the partial differential equation-based image denoising techniques, a large amount of noise remains the same as impact noise. In recent years, new models were proposed with the goal of denoising the images while preserving the edges. The smoothing filters, wavelet transform coefficients, and adaptive parameters have been used in the new models to address the drawbacks of previous approaches while at the same time improving their performance. This paper investigates the effect of a number of different parameters in diffusion-based algorithms and proposes an aggregated formula for modifying methods that enhance the image due to the power of the noise, image content, and user requirements. Furthermore, a monotonic decreasing convergence rate curve based on only the noisy image was proposed that its slope could determine the stop moment of the algorithm so that no additional influence occurred to avoid blurring. Results show that hybrid methods can have a good performance in objective and subjective quality assessment.

Index Terms—Image Processing, Image denoising, Diffusion equation, Comprehensive adaptive diffusion equation.

I. INTRODUCTION

DIGITAL images and their numerous applications such as satellite, medical, computer, and television images are significant examples of images that play an extraordinary role in human life today. During image acquisition, processing and transmission, images confront with some main challenges like the accuracy of devices, the communication path, and the natural environment. These challenges reduce image quality and almost create distortions in the observed image. Distortions include image blurring, geometric and compression distortion, and additive noise. Distorted images are not favorable to human eye observation and remarkably impair the information inside images for subsequent processing. Image enhancement is a solution to increase the sharpness and the visual effect of the image and prepare it for the future processing applications. In many image processing issues, the main step in achieving better image quality is to remove unwanted noise from the image. Various linear and nonlinear methods are presented for noise removal and image retrievals, such as neighborhood and non-local filters, using discrete cosine and wavelet transforms, singular value decomposition, bilateral and total variation filters [1], [2], [3], [4], [5], [6]. The most significant problem

in image denoising is to make the trade-off between removing noise and maintaining edges. In recent decades, methods based on variation and partial differential equations (PDEs) have gained popularity for image restoration. These methods got very successful results in eliminating noise. The principal intuitive method for removing noise from an available image is to retrieve the good estimation of original image. The noise removal technique based on anisotropic diffusion equations which were proposed in 1990 by Perona and Malik, is usually taken as a basic work on PDEs for image restoration [7]. It is commonly believed that the Perona-Malik(PM) model provides a potential algorithm for image segmentation, noise removal, edge detection, and image enhancement. For many years, many articles analyzed the capability and drawbacks of the PM method to propose modified methods [8], [9], [10], [11], [12], [13], [14]. In recent years some hybrid models based on the PM method are proposed that show remarkable results [15], [16], [17], [18], [19], [20], [21]. For a gray-scale image in continuous domain, Perona and Malik proposed an edge-controlled diffusion operator [7]:

$$\begin{aligned} \frac{\partial u}{\partial t} &= \text{div}(g(\nabla u)\nabla u), \quad (x, y, t) \in \Omega \times (0, \infty) \\ u(x, y, 0) &= f(x, y), \quad (x, y) \in \Omega \end{aligned} \quad (1)$$

where $f(x, y)$ is noisy available image, symbols div and ∇ are divergence and gradient operator with respect to spatial coordinates in a domain $\Omega \subset \mathbb{R} \times \mathbb{R}$, respectively. $u(x, y, t)$ is initialized using noisy image and the main goal is to get closer to the original image over time. $g(\nabla u)$ is a non-negative function on the magnitude of local image gradient ∇u which controls the diffusion rate and has such properties as $g(0) = 1$, and $g(\nabla u) \rightarrow 0$, as $|\nabla u| \rightarrow \infty$. Two kernel that commonly used are:

$$g(\nabla u) = \frac{1}{1 + (|\nabla u|/K)^2} \quad (2)$$

$$g(\nabla u) = \exp(-(|\nabla u|/K)^2) \quad (3)$$

where K refers to the gradient threshold of the diffusion coefficient function to control edge detection and the amount of image influence. The kernel $g(\nabla u)$ always named as Diffusion Coefficient Function (DCF). It is important note that, the lower the threshold level, the more final residual noise but less blurry, and the higher the threshold level, less the residual noise in the final result but more opaque.

In discrete domain, PM algorithm has used four directions

Fig. 1: function of diffusion amount with two threshold level, $k = 10, k = 20$ using Eq. 2 and 3.

(North, South, East, and West) to computing gradients and so it used Eq. 4 instead of Eq. 1, (note: in some cases for summarizing the formula $\mathbf{x}$ is used instead of (m, n) :

$$u^r(\mathbf{x}) - u^{r-1}(\mathbf{x}) = \tau \sum_{directions} g(\nabla_d u^{r-1}(\mathbf{x})) \cdot \nabla_d u^{r-1}(\mathbf{x});$$

$$d \in \{N, S, E, W\}$$

$$\begin{aligned} \nabla_N u^{r-1}(m, n) &= u^{r-1}(m-1, n) - u^{r-1}(m, n); \\ \nabla_S u^{r-1}(m, n) &= u^{r-1}(m+1, n) - u^{r-1}(m, n); \\ \nabla_W u^{r-1}(m, n) &= u^{r-1}(m, n-1) - u^{r-1}(m, n); \\ \nabla_E u^{r-1}(m, n) &= u^{r-1}(m, n+1) - u^{r-1}(m, n). \end{aligned} \quad (4)$$

where τ is used for regularization and always considered as $\frac{1}{4}$. The term $g(\nabla u) \cdot \nabla u$ of Eq. 4, in one direction with respect to gradient magnitude, $|\nabla u|$, is shown considering Eq. 2 and 3 in Fig. ???. As can be seen, as the magnitude of the gradient increases, the value of the function goes to zero to prevent the image from changing at important edges. There is also protection of perfectly smooth areas. However, since using Eq. 3 leads to zero, rapidly, always Eq. 2 is used.

$\nabla u^{r-1}(\mathbf{x})$ in space domain could be done in eight directions to provide more improvement. In the form of eight directions, gradient is done in Eq. 4 besides other four diagonal direction as NorthWest, NorthEast, SouthWest and SouthEast as following:

$$\begin{aligned} \nabla_{NW} u^{r-1}(m, n) &= u^{r-1}(m-1, n-1) - u^{r-1}(m, n); \\ \nabla_{NE} u^{r-1}(m, n) &= u^{r-1}(m-1, n+1) - u^{r-1}(m, n); \\ \nabla_{SW} u^{r-1}(m, n) &= u^{r-1}(m+1, n-1) - u^{r-1}(m, n); \\ \nabla_{SE} u^{r-1}(m, n) &= u^{r-1}(m+1, n+1) - u^{r-1}(m, n). \end{aligned} \quad (5)$$

which using 8 directions requires adjusting $\tau = \frac{1}{8}$. Recently, the other methods based on the fractional diffusion equation were proposed in [22], [23]. Method of [22] could be used in medical images, however, the fractional equations result

in a very complicated set of equations in multiple dimensions, which are hard to solve. The authors split diffusion equations using Alternative Direction Implicit (ADI) method that leads to be solved efficiently with the tridiagonal matrix algorithm. As well, Liao and Feng proposed the time-fractional diffusion equation to effectively remove the noise of image [23]. Their algorithm has too much computation complexity as for example, when they operate the implicit model in the image with 256×256 pixels, it costs three hours but still cannot iterate once. So, they divided an image into blocks and applying the block method. However, their numerical scheme costs 30 minutes to run five times. Therefore, we decided to investigate in section II, the important options would be defined to optimize parameters for each noisy image to restore that, tried to be able to integrate all the algorithms into one comprehensive format and analyzed with simulations in III. We concluded this paper in section IV.

II. STUDY THE KEY EFFECTIVE POINTS

What we want from the process is to achieve three main objective goals:

- Eliminate noise in image as much as possible;
- Preserve the edges of the image and prevent details from fading;
- Enhance the natural visual quality of the image.

we suggested a comprehensive formula to adjust the smoothing and noise removal according to its intensity and all optional parameters. Initially, the corrupted images is transferred to the $YCbCr$ or other color space which its brightness is in one channel and two others belongs to color components. The algorithm could be applied to three-channel independently. However, we observed for simplicity, in the case of white additive noise, it is enough efficiency to apply algorithm only to the brightness component. Other channels correspond to the color components that are retained.

The noise power of the image obtained from channel Y can be estimated using visibility or some formula such as total power of noisy image and its difference with an ordinary safe image. Due to the impact of noise, especially under heavy noise pollution, computation of the gradient is not robust to outliers. Hence noise still remains in the regions with gradients greater than the threshold level and, over-diffusion cause blurring in edges. To overcome this drawback some research works have been done especially in the recent decade. So, we started to define the key effects step by step to receive a complete formula using diffusion equation in image processing. At first, we write Eq. 6 as an initial modified formula:

$$u^r(\mathbf{x}) - u^{r-1}(\mathbf{x}) = \varphi^r(\mathbf{x}) = \tau \sum_d \psi(\nabla_d u^{r-1}(\mathbf{x})) \cdot \nabla_d u^{r-1}(\mathbf{x}) \quad (6)$$

The most important factor influencing the penetration coefficient in Eq. 6 is the function $\psi(\cdot)$, which in its simplest form is introduced in Eqs. 2 and 3 named $g(\cdot)$ in the elementary method. This function was defined as a Diffusion Coefficient Function (DCF). $\psi(\nabla u^{r-1})$ is a weighted function that allows us to change pixel values of the image. In this procedure, there

are some options to consider the effect of noise on edges and smooth areas in the image. Certainly, noise could result in the detection of mistake edges, so in the computation of DCF parameters, researchers suggested considering gradients after smoothing the degraded image to reduce the noise effects in the detection of edges. Therefore, the function $\psi(\cdot)$ could be defined generally as Eq.7.

$$\psi(\nabla u^{r-1}(\mathbf{x})) = \frac{1}{1 + \left| \frac{\nabla[u^{r-1}(\mathbf{x}) * h(\mathbf{x})]}{K} \right|^{\alpha^{r-1}}} = \frac{1}{1 + \left| \frac{u^{r-1}(\mathbf{x}) * \nabla h(\mathbf{x})}{K} \right|^{\alpha^{r-1}}} \quad (7)$$

In this equation, all options of ∇u that were explained in section I just on $u^{r-1}(\mathbf{x})$, could be applied on $u^{r-1}(\mathbf{x}) * h(\mathbf{x})$ that $h(\mathbf{x})$ is a smoothing operator. Therefore, there are several parameters to choose in the process. Obviously, considering Eq.7, in the initial PM method, this filter is replaced by the impulse function. In the following all items are explained.

A. Threshold Level K

K is the threshold level which indicates edges with a larger value are retained and the rest move smoothly under the influence of diffusion. In uniform areas where the image gradient is low and the noise is inconsistent, the DCF, $\psi(\nabla u^{r-1})$ should be greater than where the edge is. However, high-power noises may be selected and maintained as the edge by incorrectly selecting the threshold value K . So, choosing the threshold value K should be done based on the noise intensity; more intensity, larger values of K . However, there is a trade-off between the threshold value, the noise remaining, and the blur of the final result. In areas with a gradient lower than the threshold, the diffusion coefficient increases to more smoothing occur. Therefore, if the image has small details, such as a forest area, tree leaves, and or areas like zones 1-3 in Fig. 4a, they may be smoothed out in this way. Proposing values of $10 < K < 25$ based on the initial observation of the noise image and brief knowledge of its contents can be appropriate.

B. Using Smoothing Filter

In 1992, for the first time, Catte et al. proposed an improved version of the PM algorithm based on the use of a smoothing filter to ensure that noisy edges are not mistakenly placed as the main edges [11]. They used the gaussian function with suitable variance. However, the amount of smoothing can also affect the main edges. In this way, the problem of not eliminating high power noise in flat areas was somewhat corrected. This is because the amplitude of the noise is reduced by the gaussian filter and maybe reaches a smaller range than the threshold level, thus leads increasing the penetration coefficient. In 1994, the exponential filter as $h(\mathbf{x}) = \left(\frac{\beta}{2}\right) e^{-\beta|\mathbf{x}|}$ which is more accurate in detecting the edges of an image [24], was used instead of the gaussian filter, which showed better performances [12]. Fig. 2, shows the comparison of

Fig. 2: up: Comparison of (a) gaussian and exponential filters, bottom: (b) gradient of gaussian and exponential filters, ($\sigma = 1$) and ($\beta = 1$)

gaussian and exponential filters and their gradients. However, users can use different smoothing filters. Due to the low width and high amplitude in the center of the exponential filter, this eliminates low noise power and preserves more detail than the gaussian one. However, if noise intensity is suggested high since the gaussian filter is wider than the exponential one, will be a better choice for the smoothing step. It should be noted that increasing σ in gaussian and decreasing β in exponential smoother could result to excessive smoothing and blurring of the resulted image. Although in the case of very high noisy case, using large σ with gaussian filter was unavoidable.

C. Exponent variable α

The other option which can be studied is parameter α used in Eq. 7. This parameter was included in the initial algorithm by two. However, some other researchers considered $0 < \alpha \leq 1$ for entire regions and $1 \leq \alpha \leq 2$ for between areas, edges. To sharpening the edges and enhance the texture of an image, Gue *et al.* [25] proposed an adaptive PM (APM) model based on variable exponent to control the diffusion rate. Gue *et al.* used a variable α based on the edge indicator that acts differently in proportion to the regions as:

$$\alpha^r = 2 - \frac{2}{1 + \left| \frac{\nabla(u^r(\mathbf{x}) * h(\mathbf{x}))}{K_\alpha} \right|^2} \quad (8)$$

where, K_α is a new threshold value which should be larger than K . If K_α is considered smaller than K , at the very small values ∇u , coefficient α becomes saturated using Eq. 8 and remains at the same constant two as the original PM method. To describe the application conditions for different gradient values, we divide the range of gradient values into several ranges:

- $\{|\nabla u| \leq K\} \Rightarrow \text{Eq.8: } \{\alpha \rightarrow 0\} \Rightarrow \text{Eq.7: } \{\psi(\nabla u) \rightarrow 1\} \Rightarrow \text{Eq.6: } \{\varphi(\mathbf{x}) \rightarrow \nabla u\}$
- $\{K < |\nabla u| \leq K_\alpha\} \Rightarrow \text{Eq. 8: } \{0 < \alpha \leq 1\} \Rightarrow \text{Eq.7: } \{\psi(\nabla u) \leq 1\} \Rightarrow \text{Eq.6: } \{\varphi(\mathbf{x}) \leq \nabla u\}$
- $\{K_\alpha < |\nabla u|\} \Rightarrow \text{Eq.8: } \{1 < \alpha \leq 2\} \Rightarrow \text{Eq.7: } \{\psi(\nabla u) \rightarrow 0\} \Rightarrow \text{Eq.6: } \{\varphi(\mathbf{x}) \rightarrow 0\}$

Therefore, when ∇u is larger than the threshold level K_α (like in the image edge position), the variable α value tends to two, then DCF in Eq. 7 tends to zero to preserve edges

Fig. 3: Influence function, $\psi(\nabla u) \cdot \nabla u$, $K = 10$ and $K_\alpha = 20$.

with no influence. Otherwise (inside regions), which ∇u is smaller than K , α tends to zero and thus increases DCF in Eq. 7 to more diffuse to remove noise. For areas in which gradient is between K and K_α , diffusion weight $\psi(\nabla u)$ would be in the middle range. Fig. 3 shows influence function $\psi(\nabla u) \cdot \nabla u$ using the constant ($\alpha = 2$) and adaptive coefficient α , Eq. 8 with and without the use of the smoother. Since the smoother reduces the magnitude of the gradient, we suggested the effect of the smoother by multiplying the gradient value by 0.8 to simulate smoothness mode. It is clear, which in the case of smoothing, threshold levels increase by 10/8. Fig. 3 clearly shows that the use of the adaptive parameter α allows to increase the penetration rate of the areas of the image where the gradient is closer and larger than the threshold level K . The range of increment of the influence function is related directly to the value of the second threshold level K_α . Although it should be pointed that increment of the influence effect range could be resulted blurring images. So, for simplicity in simulations, we consider $K_\alpha = 2 \times K$. Obviously, designers could play with this parameter-dependent application and image contents.

To examine the effect of smoothing kernels and parameter α of the diffusion equation procedure on the recovered images, image no.25 was selected from the database TID2013 [26]. This is an artificial image that shows different paths in several directions and widths, areas with various brightness, texts, and all possible textures in the image. The models which were used here have been defined in first six rows of table I and their details in table II that $h_\sigma(\mathbf{x})$ and $h_\beta(\mathbf{x})$ are gaussian and exponential smoother respectively. By using "ssim" evaluation criteria, (Structure Similarity Index Measure) [27] the results obtained from the implementation of different models on the sample image were compared. In ssim criteria, the larger the value of criteria shows the better improvement. The destroyed image was constructed with collective the white noise such as $SNR = 16.71^{dB}$ and $ssim = 0.51$. In all simulations, $k = 15$, $k_\alpha = 30$, $\sigma = 1.5$, $\beta = 1$ were considered. Since the artificial image has been made according to certain textures, four areas of it were selected and numbered in the original image, 4a. In all the other images, four areas were magnified and below the resulted image displayed in left to right order. The images are shown in Fig. 4 corresponds to

Fig. 4: (a) Original image, (b) noisy image, the resulted images of max *ssim* in different models: (c) *ssim* = 0.6803, (d) *ssim* = 0.6717, (e) *ssim* = 0.6213, (f) *ssim* = 0.5714, (g) *ssim* = 0.5455, (h) *ssim* = 0.5194.

the peak value index *ssim*, 5a, in each model. A few key points in Fig. 4 should be noted. The values of *SNR* did not correspond in any way with the quality of the images and all of them were decreasing despite the increase in quality, Fig. 5b. All methods have an improvement process at first and then a quality reduction process due to excessive smoothing

Fig. 5: (a) *ssim* , (b) *SNR* curves refer to Figs. 4.

happened. Details of Models and the resulted images are in the followings:

- Method PM and its conformity, APM, are the slowest of all. However, the final result has more quality value *ssim* than other methods, even after 23 iterations, high noise remains like impulsive noise. APM model 4d, especially could not construct zone 4 as before.
- The EPM model is much faster than the APM, but the resulting image is somewhat blurred, especially at zones 3 and 4, Fig. 4e. EAPM model processes very faster and without blurring but with remained noise in flat areas, Fig. 4f. By repeating the algorithm, the noise is removed in uniform areas, but the thin edges are blurred at zones 2 and 3 and almost completely removed, and therefore the value of *ssim* is significantly reduced.
- In GPM model, like EPM, the noise remains in the flat areas, and in areas such as zone 1, it also appears as noise spots, Fig. 4g. The model GAPM is more opaque than the EAPM in areas with thin areas and details, and this is due to the essence of smoothing in the gaussian kernel than the exponential kernel, Fig. 4h.
- Despite the sharp decrease in the value of *ssim* in smoothing methods, the resulting shapes are completely cleared of noise, but areas with fine texture are damaged.
- Convergence speed is very attractive and remarkable in methods with smoother and adaptive parameter, as in special applications, the user can achieve the processing speed in exchange for a very low quality reduction.

To more analyze the effect of coefficient α , Figs. 6 can be considered. In these figures, the gradient images and their histograms in the north direction in models PM and EPM after five repetitions of images Figs. 4 show important notes. The range of values in the figures and the histograms are significant. In the case of without using a smoother, the range is in $[-250, 250]$, and with a smoother in $[-200, 200]$, where the reduction of the gradient value is well evident. As well, in the gradient image using smoothing, the density distribution is more concentrated around zero. Then, we calculated the value of adaptive coefficient α using Eq. 8 based on these two gradient images that lead to Figs. 6e and 6f. As seen in Fig. 6, using the adaptive α caused various values especially in flat areas which $\alpha \rightarrow 0$ and lead to more diffusion to more smoothing, specially in 6f. In the areas with too many edges, Figs. 6e and 6f, α increases to allow reduction in the influence amount to save edges.

Fig. 6: Gradient image in north direction and effect of smoothing and adaptive α : (a) gradient image Fig.4b in PM models after five iterations in north direction, (b) gradient image Fig.4b in EPM models after five iterations in north direction, (c) histogram of 6a, (d) histogram of 6b, (e) image of computed α based Eq. 8 for 6a, (f) image of computed α based Eq. 8 for 6b, (g) histogram of 6e, (h) histogram of 6f.

D. Transform Gradient

It was recommended to use a transform domain gradient along a spatial domain [15]. So, using wavelet transform that offers a simultaneous localization in the time and frequency domain, is suggested. After one level wavelet transform, three gradient image in the vertical (*V*), horizontal (*H*), and diagonal (*D*) directions which used wavelet functions and one low-pass image are provided. These three subimage *H*, *D*, and *V* play the role of ∇u in diffusion equation.

It is suggested not to use the transform gradient alone. Simultaneous use of weighted spatial and transform gradients can have better results especially in the case of the high-power noisy image. In this way, the *sym4* wavelet transform is recommended in one layer to apply the diffusion equation in each of the vertical, horizontal, and diagonal directions. We have observed that using more than one level of wavelet transform not only does not lead to better results but also increases the computational complexity. Therefore, in this step, the designer should decide about the weights of ρ_1 for spatial and ρ_2 for transform gradients that should be $\rho_1 + \rho_2 = 1$. It is advised to consider $\rho_1 = 0.75$ and $\rho_2 = 0.25$.

E. Stop Condition

The experimental results indicate the need to provide a stop condition in the algorithm just optimally state or close to it to overcome the staircase and edge blurring effects. We could stop and evaluate the resulted image to carry on or finalized the restoration procedure. In noisy cases, too many mistake edges exist that when running the algorithm, in consecutive iterations gradually disappear or smooth. Therefore, it is expected that in consecutive iterations, the changes in resulted images that are dependent on the image edges, should be reduced. According to the tips provided, we consider the average absolute value of the resulting image gradient in successive repetitions as a criterion. Specifically, since the first iteration is very productive, the average gradient relative to the first iteration as Eq. 9 is measured and compared with the threshold level SC to stop the iterations.

$$\begin{aligned} u_x^r &= \frac{\partial u^r}{\partial x}; u_y^r = \frac{\partial u^r}{\partial y}; \\ G_{u^r} &= [u_x^r]^2 + [u_y^r]^2; M^r = \text{mean}(G_{u^r}); \\ R(r) &= M^r / M^1; \\ r^* &= \min_r \{R(r) < SC\} \end{aligned} \quad (9)$$

where SC is named stop threshold condition. Experiments showed existenc dependency between noise power, image contents and the stop condition. The higher the noise power, the smaller the stop condition, and the more details in image, the larger the stop condition should be used to avoiding from blurring and degradation details. In later, we named R as the convergence rate. Note that the convergence curve is available online during processing, so, whenever it seemed sufficient, the algorithm can be finalized.

TABLE I: Introducing the acronym of different models

PM [7]	classic diffusion equation
APM [25]	classic diffusion equation with adaptive α
GPM [11]	classic diffusion equation using gaussian smoother
GAPM	diffusion equation with adaptive α using gaussian smoother filter
EPM [12]	classic diffusion equation using exponential smoother
EAPM	diffusion equation with adaptive α using exponential smoother
WPM [15]	combined diffusion Equation of spatial and wavelet
WAPM	combined diffusion Equation of spatial and wavelet with adaptive α
WGPM	combined diffusion Equation of spatial and wavelet using gaussian smoother
WGAPM	combined diffusion Equation of spatial and wavelet with adaptive α using Gaussian smoother
WEPM	combined diffusion Equation of spatial and wavelet using exponential smoother
WEAPM	combined diffusion Equation of spatial and wavelet with adaptive α using exponential smoother

F. Comprehensive diffusion Equation

Due to the points mentioned, the restoration complete integrated formula is presented as follows:

$$\begin{aligned} u^0 &= \text{initial noisy image} \\ u^r(\mathbf{x}) - u^{r-1}(\mathbf{x}) &= \varphi^r(\mathbf{x}) = \\ \rho_1 + \rho_2 &= 1; \quad \rho_1 \varphi_S^r(\mathbf{x}) + \rho_2 \varphi_W^r(\mathbf{x}) = \\ \varphi_j^r(\mathbf{x}) &= \tau_j \sum_d \psi \left(\nabla_d^j u^{r-1}(\mathbf{x}) \right) \cdot \nabla_d^j u^{r-1}(\mathbf{x}) \\ \text{if } j = S &\left\{ \begin{aligned} d \in \{N, S, E, W, SW, EW, NW, NE\}; \\ \psi(\nabla u^r(\mathbf{x})) &= \frac{1}{1 + \left(\frac{|\nabla u^r(\mathbf{x}) * h(\mathbf{x})|}{k} \right)^{\alpha^r}} \\ \alpha^r &= 2 - \frac{2}{1 + \left| \frac{\nabla u^r(\mathbf{x}) * h(\mathbf{x})}{k_\alpha} \right|^2} \end{aligned} \right. \quad (10) \\ \text{if } j = W &\left\{ \begin{aligned} d \in \{H, V, D\}; \\ \psi(\nabla u^r(\mathbf{x})) &= \frac{1}{1 + \left(\frac{|\nabla u^r(\mathbf{x})|}{k} \right)^{\alpha^r}} \\ \alpha^r &= 2 - \frac{2}{1 + \left| \frac{\nabla u^r(\mathbf{x})}{k_\alpha} \right|^2} \end{aligned} \right. \end{aligned}$$

where $\nabla^S u^{r-1}(\mathbf{x})$ and $\nabla^W u^{r-1}(\mathbf{x})$ are respectively gradient in space domain and transform wavelet domain. The combined equation makes it possible to use all or part of the available options. The transform gradient can be used along with the smoothness and the adaptive α . Tables I and II consider all possible cases. Chosen models to apply the diffusion algorithm is a dependent procedure on the noise intensity and the essence of the image.

Some other modified algorithms were proposed to overcome the edge blurring and staircase effect in the PM diffusion model. For example, in [13] fourth-order PDE for image denoising is designed. Although the fourth-order PDE is free of staircase effect, it often suffers over-smoothing, and the calculation of a high-order PDE is complicated and challenging. In next section, all models were experimented with two ordinary images.

III. EXPERIMENT RESULTS AND COMPARISONS

All experimented models have been summerized with abbreation names in table I and using vital equations in table II.

Although we tested a large number of images, the results of two images are only displayed. These images were selected with different contexts from TID2013 dataset [26], which can

TABLE II: Essential equations in models of table I, h_σ and h_β are gaussian and exponential smoother respectively.

PM [7]	$\psi = \frac{1}{1 + \left \frac{\nabla_d^S u}{k} \right ^2}$
APM [25]	$\psi = \frac{1}{1 + \left \frac{\nabla_d^S u}{k} \right ^\alpha}; \alpha = 2 - \frac{2}{1 + \left \frac{\nabla_d^S(u)}{k_\alpha} \right ^2}$
GPM [11]	$\psi = \frac{1}{1 + \left \frac{\nabla_d^S(u * h_\sigma)}{k} \right ^2}$
EPM [12]	$\psi = \frac{1}{1 + \left \frac{\nabla_d^S(u * h_\beta)}{k} \right ^2}$
GAPM	$\psi = \frac{1}{1 + \left \frac{\nabla_d^S(u * h_\sigma)}{k} \right ^\alpha}; \alpha = 2 - \frac{2}{1 + \left \frac{\nabla_d^S(u * h_\sigma)}{k_\alpha} \right ^2}$
EAPM	$\psi = \frac{1}{1 + \left \frac{\nabla_d^S(u * h_\beta)}{k} \right ^\alpha}; \alpha = 2 - \frac{2}{1 + \left \frac{\nabla_d^S(u * h_\beta)}{k_\alpha} \right ^2}$
WPM [15]	$\psi_d^S = \frac{1}{1 + \left \frac{\nabla_d^S u}{k} \right ^2}; \psi_d^W = \frac{1}{1 + \left \frac{\nabla_d^W u}{k} \right ^2}$ $\psi = 0.75\psi_d^S + 0.25\psi_d^W$
WAPM	$\psi_d^S = \frac{1}{1 + \left \frac{\nabla_d^S(u)}{k} \right ^\alpha}; \alpha_s = 2 - \frac{2}{1 + \left \frac{\nabla_d^S(u)}{k_\alpha} \right ^2}$ $\psi_d^W = \frac{1}{1 + \left \frac{\nabla_d^W u}{k} \right ^\alpha}; \alpha_w = 2 - \frac{2}{1 + \left \frac{\nabla_d^W(u)}{k_\alpha} \right ^2}$ $\psi = 0.75\psi_d^S + 0.25\psi_d^W$
WGPM	$\psi_d^S = \frac{1}{1 + \left \frac{\nabla_d^S(u * h_\sigma)}{k} \right ^\alpha}; \psi_d^W = \frac{1}{1 + \left \frac{\nabla_d^W u}{k} \right ^\alpha};$ $\psi = 0.75\psi_d^S + 0.25\psi_d^W$
WGAPM	$\psi_d^S = \frac{1}{1 + \left \frac{\nabla_d^S(u * h_\sigma)}{k} \right ^\alpha}; \alpha_s = 2 - \frac{2}{1 + \left \frac{\nabla_d^S(u * h_\sigma)}{k_\alpha} \right ^2}$ $\psi_d^W = \frac{1}{1 + \left \frac{\nabla_d^W u}{k} \right ^\alpha}; \alpha_w = 2 - \frac{2}{1 + \left \frac{\nabla_d^W(u)}{k_\alpha} \right ^2}$ $\psi = 0.75\psi_d^S + 0.25\psi_d^W$
WEPM	$\psi_d^S = \frac{1}{1 + \left \frac{\nabla_d^S(u * h_\beta)}{k} \right ^\alpha}; \psi_d^W = \frac{1}{1 + \left \frac{\nabla_d^W u}{k} \right ^\alpha};$ $\psi = 0.75\psi_d^S + 0.25\psi_d^W$
WEAPM	$\psi_d^S = \frac{1}{1 + \left \frac{\nabla_d^S(u * h_\beta)}{k} \right ^\alpha}; \alpha_s = 2 - \frac{2}{1 + \left \frac{\nabla_d^S(u * h_\beta)}{k_\alpha} \right ^2}$ $\psi_d^W = \frac{1}{1 + \left \frac{\nabla_d^W u}{k} \right ^\alpha}; \alpha_w = 2 - \frac{2}{1 + \left \frac{\nabla_d^W(u)}{k_\alpha} \right ^2}$ $\psi = 0.75\psi_d^S + 0.25\psi_d^W$

TABLE III: SNR and $ssim$ of noisy images used for representation.

image	SNR	$SSIM$
House	16.35 ^{dB}	0.3993
Lena	17.33 ^{dB}	0.1708
Parrots	17.19 ^{dB}	0.1276

Fig. 7: (a) and (b) original and noisy House image, (c) and (d) original and noisy Lena image, (e) and (f) original and noisy Parrots image.

be seen in Fig. 7. The selected images are in two different categories.

- 1. Image House is complex and has many distinguishable edges in different directions and also some flat areas;
- 2. Image Lena consider as a medium complexity image.
- 3. Image Parrots has two prominent objects and the background is almost uniform, the noise of which is obvious to the viewer, but its blur is not annoying.

To show the performance of methods, we provided noisy case for each image as table III which can be seen in Fig. 7. Although all three images are the same in terms of the noise applied to them (quantity SNR is almost equal), quantity $ssim$ is very different, which is the result of the type of image structure and the possibility of seeing noise in them. Reconstructed images from different models were compared using two evaluation criteria. The first one is $ssim$ and the second one was introduced in 2018 [28] as QPSV based on singular value decomposition, the structure of images, and salience of image

parts. Unlike $ssim$, the lower the value of QPSV shows the better improvement that bounded with zero. Because in this image reconstruction procedure, the goal is to reduce noise for better viewing of images, SNR criterion, which calculates the point-to-point difference, does not provide appropriate values for comparison. For this purpose, we showed the results of two criteria $ssim$ and QPSV. The results of both criteria mentioned in the procedure with 10 consecutive repetitions for three noisy images were shown in Figures 8, 9. The convergence of all experiments are shown in Figs. 10. The situation where the stop condition was satisfied, indicated by a mark $\square$. The stop condition was set $SC = 0.6$ for House and 0.5 for Parrots and Lena images, respectively. The choice of stop condition value depends on the user and the application of the image. In these experiments, the stop condition value was selected despite having approximately maximum in curves $ssim$ to see their effects on the convergence curves. Specifically, when the slope of the convergence curve changes and approaches zero,

Fig. 8: *SSIM* plots of three selected noisy images in consecutive iterations, (a) House, (b) Lena and (c) Parrots pictures.

it is the stop time. As mentioned, this value depends on the image structure, the crowds, the conceptual nature, and the uniform areas. In all simulations, $K = 15$, $K_\alpha = 30$, $\sigma = 1.5$ for gaussian filter and $\beta = 1$ for exponential filter were considered. The time required to perform each iteration in model WEAPM, which uses all possible options using cpu 3.7^{GHz} was 0.38 seconds for image Lena in size 512×512 and 0.22 seconds for images House and Parrots in size 512×384 .

As mentioned earlier, the visual results and evaluation qualities for ordinary images are slightly different from those in Fig 4. Some comments on Figs. 8 to 13 are shown:

- it is clear that the use of a smoother, regardless of its type, will lead to a faster reconstruction and better quality. The figures show the relative impact of the smoother on the other parameters (adaptive α and transform gradient). As mentioned, in images with high noise, the use of gaussian smoothing filters accelerated the achievement of good quality results. Please look at Figs. 8 to 10 in red and green curves that correspond to using gaussian and exponential smoothers, respectively.

Fig. 9: *QPSV* plots of three selected noisy images in consecutive iterations, (a) House, (b) Lena and (c) Parrots pictures.

- In general, using the adaptative α will help accelerate convergence, especially in the absence of a smoother and is more pronounced if we are dealing with high noisy images.
- The effect of using a wavelet transform alone was not effective, but when combined with a smoother. The effect of its improvement is more than using the adaptive parameter. This may be because the use of wavelet transform with wave 'sym4' actually increases smoothing.
- In the case of smoother methods, which operate faster, the initial rapid convergence caused to receiving stop condition *SC* earlier than other methods. In continuous repetitions, infiltration changes gradually subside, although they could still provide better image quality.
- It is evident and significant that convergence rate, Eq. 9 is only dependent on observed available image and independent from the original image. The convergence rate R is the only tool that we have during the processing and must pay maximum attention to its curve and properties. As shown in Figs. 10, monotonic decreasing for all methods guarantee the stability of methods. We saw

Fig. 10: Convergence rate based on Eq. 9 for three selected noisy images in consecutive iterations, (a) House, (b) Lena and (c) Parrots pictures.

that the curve slope in the first iteration is very sharp and after some iterations, the slope of the curve slows down and approaches to zero. After the iteration in which the slope rate curve starts to slow down, dependent on the application, the algorithm could be stopped. After the stop point, image reconstruction usually tends to blur the image. This feature has less effect on image quality in images with more uniform areas, but reduces the quality of vision in complex images with details as seen in Fig. 8a.

- The stop condition depends on both the nature of the image and the noise power. For example, as seen in the curves of Figs. 10, the stop condition for House picture was 0.6, but for the images of Parrots and Lena, should be decreased to 0.5 to stop in the best time. However, the number of iterations are not much conflict. It happens because the stop condition is based on resulted image gradient that is related to image contents. In images with more details, SC should be larger than images that have uniform areas.

Fig. 11: Resulted restored images refer to Fig.7b noisy house image which iteration that SC was satisfied, iteration number is in (.).

- Criterion QPSV shows the convergence of the algorithm better than $ssim$. As well as, QPSV curves are more in line with convergence rate curves, Figs. 9 and 10. However, QPSV curves are plotted using reference original images, convergence rate curves are plotted without reference and can be used in real applications. The main improvement is in the initial repetition stages that after reaching the condition of stopping, the changes become minor.

As a final test, we used Lena picture with very little ($SNR = 22.5^{dB}$) and very much noise ($SNR = 12.9^{dB}$), and the results of the quality value $ssim$ and the convergence rate of the simulation with all the models in Table II are shown in Figs. 14. As it is seen, in the case of high noise, it is possible to increase $ssim$ from about 0.05 to 2.5, and all models eventually reach the same state, but the speed of receiving their optimal state is different. For a low-noise image, it is clear that using a smoother is not helpful (in contrast to the high

Fig. 12: Resulted restored images refer to Fig.7d noisy Lena image which iteration that SC was satisfied, iteration number is in (.).

Fig. 13: Resulted restored images refer to Fig.7f noisy Parrot image which iteration that SC was satisfied, iteration number is in (.).

noisy case), especially after three-four steps in which a good image is obtained, the resulting images tend to blur, which in models using a smoother, this state is done with more slope. The best image is established by APM and the least blurring is provided by WAPM at the end of 10 repetitions. Figs. 15 show some reconstructed images. Shortly, as mentioned, by playing with different values and choices, appropriate results can be achieved with the application. The comprehensive model, Eq. 10 could be introduced as the simplest and the most efficient in image denoising based diffusion equation.

IV. CONCLUSIONS

In this paper, a comprehensive overview of the various models of image denoising using the diffusion equation was provided. The gaussian and exponential filters as a pre-processing stage of the diffusion process were evaluated and compared. We investigated adaptive parameter α in the diffusion equation and found to be attractive especially in

Fig. 14: (a) $ssim$ of the resulted low noisy Lena image, (b) convergence rate of the resulted of low noisy Lena image, (c) $ssim$ of the resulted high noisy Lena image, (d) convergence rate of the resulted of high noisy Lena image,

absence of smoothing operator. It must move slowly from zero to two with respect to increasing the gradient value. To improve the infiltration process especially in high power noise, using the smoothing filter was vital in denoising. The gradient calculation in the spatial domain along in the wavelet domain is suggested in the case of high noisy images. The combined algorithm can therefore be used for images with different backgrounds (smooth, semi-smooth, crowded). Considering the stop condition can also stop the iterations close to the optimal state. The convergence rate R is independent from the original image that is the only tool during the processing to controlling in real applications. Because of the existing high correlation between R and QPSV quality measurement, convergence rate could be used as no reference image quality measurement at runtime procedure. Observation results show that smoothing filters and adaptive α can achieve asymptotic quality with the increasing speed of processing.

REFERENCES

- [1] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 2. IEEE, 2005, pp. 60–65.
- [2] —, "Non-local means denoising," *Image Processing On Line*, vol. 1, pp. 208–212, 2011.
- [3] Z. Kai, L. Hong, Y. Shuyuan, W. Min, L. Zhengkang, Q. Rong, and J. Licheng, "Sparse double-geometric nonlocal mean image recovery via steerable kernel," *Applied Soft Computing*, vol. 33, pp. 77–85, 2015.
- [4] M. Grasmair, "Locally adaptive total variation regularization," in *International Conference on Scale Space and Variational Methods in Computer Vision*. Springer, 2009, pp. 331–342.
- [5] H. Liu, R. Xiong, X. Zhang, Y. Zhang, S. Ma, and W. Gao, "Nonlocal gradient sparsity regularization for image restoration," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 9, pp. 1909–1921, 2016.

Fig. 15: (a) low noisy Lena, (b) high noisy Lena, (c) and (e) resulted restoring low noisy image using method and iteration number is in (.), (d) and (f) resulted restoring high noisy image using method and iteration number is in (.), .

- [6] M. Gharbi, J. Chen, J. Barron, S. Hasinoff, and F. Durand, "Deep bilateral learning for real-time image enhancement," *ACM Transactions on Graphics*, vol. 36, no. 4, pp. 1–12, 2017.
- [7] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 12, no. 7, pp. 629–639, 1990.
- [8] M. J. Black, G. Sapiro, D. H. Marimont, and D. Heeger, "Robust anisotropic diffusion," *IEEE Transactions on image processing*, vol. 7, no. 3, pp. 421–432, 1998.
- [9] J. Weickert, *Anisotropic diffusion in image processing*. Teubner Stuttgart, 1998, vol. 1.
- [10] Y.-L. You, W. Xu, A. Tannenbaum, and M. Kaveh, "Behavioral analysis of anisotropic diffusion in image processing," *IEEE Transactions on Image Processing*, vol. 5, no. 11, pp. 1539–1553, 1996.
- [11] F. Catté, P.-L. Lions, J.-M. Morel, and T. Coll, "Image selective smoothing and edge detection by nonlinear diffusion," *SIAM Journal on Numerical analysis*, vol. 29, no. 1, pp. 182–193, 1992.
- [12] F. Torkamani-Azar and K. Tait, "Image recovery using the anisotropic diffusion equation," *IEEE Transactions on Image Processing*, vol. 5, no. 11, pp. 1573–1578, 1996.
- [13] Y.-L. You and M. Kaveh, "Fourth-order partial differential equations for noise removal," *IEEE Transactions on Image Processing*, vol. 9, no. 10, pp. 1723–1730, 2000.
- [14] G. Gilboa, N. Sochen, and Y. Y. Zeevi, "Forward-and-backward diffu-

- sion processes for adaptive image enhancement and denoising," *IEEE transactions on image processing*, vol. 11, no. 7, pp. 689–703, 2002.
- [15] J. Liu, K. She, Y. Li, and Q. Tu, "Image diffusion filtering algorithm combined with variable exponent and convective constraint," *Signal, Image and Video Processing*, vol. 13, no. 1, pp. 87–94, 2019.
 - [16] N. Wang, Y. Shang, Y. Chen, M. Yang, Q. Zhang, Y. Liu, and Z. Gui, "A hybrid model for image denoising combining modified isotropic diffusion model and modified perona-malik model," *IEEE Access*, vol. 6, pp. 33 568–33 582, 2018.
 - [17] T. Barbu, "A hybrid nonlinear fourth-order pde-based image restoration approach," in *2016 20th International Conference on System Theory, Control and Computing (ICSTCC)*. IEEE, 2016, pp. 761–765.
 - [18] Y. Chen, Y. Bai, Q. Zhang, Y. Wang, and Z. Gui, "Self-adaptive anisotropic image enhancement algorithm based on local variance," *Journal of Engineering Science & Technology Review*, vol. 10, no. 3, 2017.
 - [19] L. Deng, H. Zhu, Z. Yang, and Y. Li, "Hessian matrix-based fourth-order anisotropic diffusion filter for image denoising," *Optics & Laser Technology*, vol. 110, pp. 184–190, 2019.
 - [20] H. K. Rafsanjani, M. H. Sedaaghi, and S. Saryazdi, "Efficient diffusion coefficient for image denoising," *Computers & Mathematics with Applications*, vol. 72, no. 4, pp. 893–903, 2016.
 - [21] A. A. Yahya, J. Tan, and M. Hu, "A blending method based on partial differential equations for image denoising," *Multimedia tools and applications*, vol. 73, no. 3, pp. 1843–1862, 2014.
 - [22] A. Abirami, P. Prakash, and K. Thangavel, "Fractional diffusion equation-based image denoising model using cngl scheme," *International Journal of Computer Mathematics*, vol. 95, no. 6-7, pp. 1222–1239, 2018.
 - [23] X. Liao and M. Feng, "Time-fractional diffusion equation-based image denoising model," *Nonlinear Dynamics*, vol. 103, no. 2, pp. 1999–2017, 2021.
 - [24] J. Shen and S. Castan, "An optimal linear operator for step edge detection," *CVGIP: Graphical models and image processing*, vol. 54, no. 2, pp. 112–133, 1992.
 - [25] Z. Guo, J. Sun, D. Zhang, and B. Wu, "Adaptive perona-malik model based on the variable exponent for image denoising," *IEEE Transactions on Image Processing*, vol. 21, no. 3, pp. 958–967, 2011.
 - [26] N. Ponomarenko, L. Jin, O. Ieremeiev, V. Lukin, K. Egiazarian, J. Astola, B. Vozel, K. Chehdi, M. Carli, F. Battisti *et al.*, "Image database TID2013: Peculiarities, results and perspectives," *Signal processing: Image communication*, vol. 30, pp. 57–77, 2015.
 - [27] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.
 - [28] F. Torkamani-Azar and J. Parkkinen, "Image quality assessment using block-based weighted svd," *Signal, Image and Video Processing*, vol. 12, no. 7, pp. 1337–1344, 2018.