Solving Heat Flow Equation for Image Regularization

Bekir Dizdaroglu
Department of Computer Engineering, Karadeniz Technical University, Trabzon, Turkey
bekir@ktu.edu.tr

ABSTRACT – In this paper, we investigated solutions of some partial differential equations (PDEs) using both isotropic diffusion and the Fourier transform for denoising grayscale images. Nowadays, there are a number of methods to solve digital image processing problems, which employ mathematical algorithms. Isotropic diffusion, acting as a linear filter, is the most basic approach to regularize or smooth image data; however, it blurs fine details of images. Until now, we have been interested in diffusion or heat equation and the convolution of the given image with the Gaussian kernel at a specific time. Experimental results show that both approaches generate almost similar outputs.

Keywords: image denoising, isotropic smoothing, Fourier transform, heat equation

I. INTRODUCTION

In image processing, researchers have been quite interested in PDE-based frameworks in recent years. There has been a lot of research based on PDEs concerning image processing. In addition, the Fourier transform with convolution is commonly used to solve PDE-based problems. In fact, the Fourier transform is essential to handle some PDEs.

Tikhonov [1] presented an isotropic smoothing method to regularize digital images. Tschumperlé et al. [2] proposed the trace-based PDE method to regularize multi-valued images. However, this method was not very successful due to the fact that it rounded corners in images. Then, Tschumperlé resolved that problem by creating an algorithm based on curvature-preserving PDE [3] and a patch-based anisotropic diffusion PDE [4]. Dizdaroglu introduced fuzzy-based anisotropic smoothing methods to remove noise from grayscale images [5]-[6]. Osgood [7] presented some basic approaches to solve differential equations in the Fourier or frequency domain. However, these methods are not explained in details.

We worked on regularization problems in many applications in digital image processing. We gave basic mathematical approaches to solve the heat flow equation using two frameworks. The first was PDE, whose solution is an isotropic smoothing. The second was the Fourier transform, whose solution is a convolution of the noisy image with the Gaussian kernel at a particular time.

II. IMAGE SMOOTHING

Let $u: \Omega \rightarrow \mathbb{R}$ be a grayscale image defined on the domain of $\Omega \rightarrow \mathbb{R}^2$. The smoothing process is simply to regularize image data. For this purpose, several mathematical methods have been improved recently. In the following subsections, two approaches, which are basic regularization methods in the digital image restoration field, are explained in detail.

II. Tikhonov Regularization Method

The Tikhonov regularization that is used in the smoothing process is known as a standard linear filtering approach. This method minimizes the variation of image $u$ by estimating the gradient norm $\|\nabla u\|$:

$$\min_{u \in L^2} E(u) = \int_{\Omega} \|\nabla u\|^2 d\Omega$$

(1)

In (1), the image gradient denoted by $\nabla u$ is the derivation of grayscale image $u$ with respect to its spatial coordinates $x = (x, y)$ (see Figure 1):

$$\nabla u = (u_x, u_y)^T = \left(\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}\right)^T$$

(2)

A vector $\nabla u: \Omega \rightarrow \mathbb{R}^2$ is defined to hold magnitudes of the grayscale image $u$ and its maximum variation directions. For image analysis, a scalar and pointwise measure of image variations, which is given by the gradient norm $\|\nabla u\|$, is defined as follows (see Figure 2):

$$\|\nabla u\| = \sqrt{u_x^2 + u_y^2}$$

(3)
where \( u_x \) and \( u_y \) are first derivatives of the image \( u \) in the \( x \) and \( y \) directions, respectively, and are calculated by Taylor's formula:

\[
\begin{align*}
    u(x + 1, y) &= u(x, y) + u_x(x, y) \\
    u(x, y + 1) &= u(x, y) + u_y(x, y)
\end{align*}
\]

If (4) is solved by using the forward finite differences method,

\[
\begin{align*}
    u_x(x, y) &= u(x + 1, y) - u(x, y) \\
    u_y(x, y) &= u(x, y + 1) - u(x, y)
\end{align*}
\]

are obtained.

![Gradient vector \( \nabla u \) at point \( x \).](image1)

**Figure 1** – Gradient vector \( \nabla u \) at point \( x \).

**Figure 2** – Gradient norm of test image of Lena (head region).

Figuring out the function \( u \), which minimizes the functional \( E(u) \), takes complex processing. However, thanks to the Euler-Lagrange equations, a minimum of \( E(u) \) can be reached easily:

\[
\frac{\partial E}{\partial u} = \frac{\partial F}{\partial u} - \frac{d}{dx} \frac{\partial F}{\partial u_x} - \frac{d}{dy} \frac{\partial F}{\partial u_y} = 0
\]

where \( F = \| \nabla u \|^2 = u_x^2 + u_y^2 \).

\[
\frac{d}{dx} \frac{\partial F}{\partial u_x} \quad \text{and} \quad \frac{d}{dy} \frac{\partial F}{\partial u_y}
\]

are calculated by using the standard differentiation rules as follows:

\[
\begin{align*}
    \frac{\partial F}{\partial u_x} &= \frac{\partial}{\partial u_x} \left( (u_x^2 + u_y^2)^{\frac{1}{2}} \right) = 2u_x \\
    \frac{\partial F}{\partial u_x} &= \frac{d}{dx} \frac{\partial F}{\partial u_x} = \frac{d}{dx} (2u_x) = 2u_{xx} \\
    \frac{\partial F}{\partial u_y} &= \frac{d}{dy} \frac{\partial F}{\partial u_y} = 2u_{yy}
\end{align*}
\]

A standard iterative method called gradient descent is used to solve the equation in (1). Actually, (1) can be seen as the n gradient of the functional \( E(u) \). A local minimizer \( u_{\text{min}} \) of \( E(u) \) can be found by starting from \( u_0 = u_{\text{noisy}} \) and then following the opposite direction of the gradient:

\[
\frac{u(x+1,y)}{\partial t} = u_0 \left( \frac{\partial F}{\partial u} - \frac{d}{dx} \frac{\partial F}{\partial u_x} - \frac{d}{dy} \frac{\partial F}{\partial u_y} \right)
\]

By solving (8),

\[
\frac{\partial u}{\partial t} = u_{xx} + u_{yy} = \Delta u
\]

is reached. Here, \( \Delta \) is the Laplace operator. This equation is called a heat flow (diffusion) equation. To find the second derivative \( u_{xx} \), Taylor's formula is used:

\[
\begin{align*}
    u(x + 1, y) &= u(x, y) + u_x(x, y) + \frac{u_{xx}(x, y)}{2} \\
    u(x - 1, y) &= u(x, y) - u_x(x, y) + \frac{u_{xx}(x, y)}{2}
\end{align*}
\]

As a result, (11) is derived from (10) by summation, and \( u_{yy} \) is calculated similarly.

\[
\begin{align*}
    u_{xx} &= u(x + 1, y) + u(x - 1, y) - 2u(x, y) \\
    u_{yy} &= u(x, y + 1) + u(x, y - 1) - 2u(x, y)
\end{align*}
\]

**U. Fourier Transform Approach**

Assume that a circle is heated up without even distribution. Heat flows around the circle and temperature changes in the course of time [7].

Let \( u(t) \), \( 0 \leq t \leq 1 \), be the temperature as a function of spatial position at a particular time \( t \). The heat equation is \( u_t = \frac{\partial u}{\partial t} = \frac{1}{2} u_{xx} \) as can be found in (9), but the 1-D heat equation is taken into account here. Assume that the function \( u(t) \) is periodic in the variable \( x \) (its period is 1), e.g., \( u(t)(x + 1) = u(t)(x) \). \( u(t) \) is defined as Fourier series with coefficients at a particular time \( t \):

\[
\begin{align*}
    u(t)(x) &= \sum_{n=-\infty}^{\infty} c_n(t)e^{2\piinx} \\
    c_n(t) &= \int_0^1 e^{-2\piinx} u(t)(x) dx
\end{align*}
\]

and \( c_n(t) \) is differentiated with respect to time \( t \) as follows:

\[
\frac{\partial c_n(t)}{\partial t} = \int_0^1 u(t)e^{-2\piinx} dx
\]
Using $u_t = \frac{1}{2} u_{xx}$, (13) is rewritten as:

$$\frac{\partial c_n(t)}{\partial t} = \int_0^1 \frac{1}{2} u_{xx} e^{-2\pi i nx} dx$$

(14)

Then, the following equation is obtained:

$$\begin{align*}
\frac{\partial c_n(t)}{\partial t} &= \int_0^1 \frac{1}{2} u_t \frac{d^2}{dx^2} e^{-2\pi i nx} dx \\
&= \int_0^1 \frac{1}{2} u_t (-4\pi^2 n^2) e^{-2\pi i nx} dx \\
&= -2\pi^2 n^2 \int_0^1 u_t e^{-2\pi i nx} dx \\
&= -2\pi^2 n^2 c_n(t)
\end{align*}$$

(15)

c_n(t) = c_n(0) e^{-2\pi^2 n^2 t} \text{ is obtained from (15).}

At time $t = 0$, the temperature $u(0)$ is denoted by the initial function $u_0$:

$$u(0) = u_0, u_0(x + 1) = u_0(x), \forall x \in \mathbb{R}$$

(16)

and the integral representation is for $c_n(0)$

$$c_n(0) = \int_0^1 u_0(x) e^{-2\pi i nx} dx$$

(17)

$$\begin{align*}
= \int_0^1 u_0(x) e^{-2\pi i nx} dx &= \hat{u}_0(n)
\end{align*}$$

where $\hat{u}_0$ is the $n^{th}$ Fourier coefficient of $u_0$. In this way, the following equation is reached:

$$c_n(t) = \hat{u}_0(n) e^{-2\pi^2 n^2 t}$$

(18)

The 1-D heat equation

$$u(t,x) = \sum_{n=-\infty}^{\infty} \hat{u}_0(n) e^{-2\pi^2 n^2 t} e^{2\pi i nx}$$

(19)

is found from (12).

For $t = 0$, the following equation is obtained from (19):

$$u(0)(x) = \sum_{n=-\infty}^{\infty} \hat{u}_0(n) e^{-2\pi^2 n^2 0} e^{2\pi i nx}$$

$$= \sum_{n=-\infty}^{\infty} \hat{u}_0(n) e^{2\pi i nx}$$

(20)

And the Fourier series for $u_0(x)$ is

$$u(0)(x) = \sum_{n=-\infty}^{\infty} \hat{u}_0(n) e^{2\pi i nx} = u_0(x)$$

(21)

Also, for $t = \infty$, the following equation is found:

$$\lim_{t \to \infty} u(t)(x) = \hat{u}_0(0)$$

(22)

is equal to the following expression:

$$\hat{u}_0(0) = \int_0^1 u_0(x) dx$$

(23)

which is the average of the initial temperature, $u_0(x)$. Then, for $\hat{u}_0(n)$, the formula is rewritten as:

$$\hat{u}_0(n) = \int_0^1 u_0(p) e^{-2\pi i np} dp$$

(24)

As a result, $u(t)(x)$ is calculated as:

$$u(t)(x) = \sum_{n=-\infty}^{\infty} e^{-2\pi^2 n^2 t} e^{2\pi i nx} \int_0^1 u_0(p) e^{-2\pi i np} dp$$

(25)

$$= \int_0^1 \sum_{n=-\infty}^{\infty} e^{-2\pi^2 n^2 t} e^{2\pi i n(x-p)} u_0(p) dp$$

Replacing the integral with (26)

$$g(t)(x-p) = \sum_{n=-\infty}^{\infty} e^{-2\pi^2 n^2 t} e^{2\pi i n(x-p)}$$

(26)

The final equation is obtained as:

$$u(t)(x) = \int_0^1 g(t)(x-p) u_0(p) dp$$

(27)

(27) is the convolution of $u_0$ and $g$.

On the other hand, the Fourier transform, $\mathcal{F}$, of the derivative of $u(s)$ is:
\[ \mathcal{F} \frac{du(s)}{ds} = 2\pi is \mathcal{F} u(s) \] \hspace{1cm} (28)

(28) is written for \( n \)th derivatives as follows:

\[ \mathcal{F} \left[ \frac{d^n u(s)}{ds^n} \right] = (2\pi is)^n \mathcal{F} [u(s)] \] \hspace{1cm} (29)

Let the Fourier transform of both sides of the heat equation with respect to \( x \) be calculated. The Fourier transform of the right-hand side of the equation, \( \frac{1}{2} \Delta u_{xx} \), found from (29), is the following:

\[ \frac{1}{2} \mathcal{F} [u_{xx}] = -2\pi^2 s^2 \mathcal{F} [u_{(t)}(s)] \] \hspace{1cm} (30)

Thus, taking the Fourier transform of both sides of the equation, the following equation is reached:

\[ \frac{\partial \mathcal{F} [u_{(t)}(s)]}{\partial t} = -2\pi^2 s^2 \mathcal{F} [u_{(t)}(s)] \] \hspace{1cm} (31)

This is a differential equation in \( t \). Then, the following equation is reached:

\[ \mathcal{F} [u_{(t)}(s)] = \mathcal{F} [u_0(s)] e^{-2\pi^2 s^2 t} \] \hspace{1cm} (32)

In (32), the exponential term on the right-hand side is the Fourier transform of the Gaussian kernel:

\[ g_{(t)}(x) = \frac{1}{\sqrt{4\pi t}} e^{-x^2/2t} \] \hspace{1cm} (33)

(32) is rewritten as product of two Fourier transforms:

\[ \mathcal{F} [u_{(t)}(s)] = \mathcal{F} [g_{(t)}(s)] \mathcal{F} [u_0(s)] \] \hspace{1cm} (34)

Basically, a convolution equation is obtained in the spatial domain:

\[ u_{(t)}(x) = g_{(t)}(x) * u_0(x) \]

\[ = \int \frac{1}{\sqrt{2\pi t}} e^{-|x-p|^2/2t} u_0(p) dp \] \hspace{1cm} (35)

If we extend (35) for 2-D, we obtain the following equations:

\[ \int g_{(t)}(x-p, y-q) u_0(p, q) dp dq \]

or

\[ \int g_{(t)}(p, q) u_0(x-p, y-q) dp dq \] \hspace{1cm} (36)

As a result, (9) is equal to the following expression, given the convolution of \( u_0 \) with a normalized 2-D Gaussian kernel \( G_\sigma \) with a variance of \( \sigma = \sqrt{\frac{1}{2t}} \) at a particular time \( t \) [2]-[3]:

\[ u_{(t)} = u_0 * G_\sigma \] \hspace{1cm} (37)

e.g.,

\[ u_{(t)}(x,y) = \int u_0(x-p,y-q) G_\sigma(p,q) dp dq \] \hspace{1cm} (38)

means the linear smoothing in 2-D, where \( G_\sigma = \frac{1}{4\pi t} \exp \left( -\frac{p^2 + q^2}{4t} \right) \).

III. EXPERIMENTAL RESULTS

As can be seen in Figure 3.a-b, an artificial Gaussian noise (\( \sigma = 20 \)) is added to the grayscale image of Lena in size of 512 \times 512 pixels for tests. The methods are also applied on an originally degraded image in size of 342 \times 259 pixels for denoising process as shown in Figure 4.a.

The output images obtained by isotropic smoothing approach and the convolution method are shown in Figure 3.c-d. Moreover, the peak signal-to-noise ratio (PSNR) between the original image and restored images are yielded. Also, the smoothing results of originally degraded image are shown in Figure 4.b-c. The methods are presenting almost the same performance compared with each other.

Related methods act as low-pass filters which suppress high frequencies in the image. Unfortunately, due to the fact that image edges and noise contain both high frequency signals, edges are quickly blurred by these approaches.
Figure 3 – Chosen parts of images of Lena for denoising: (a) the original image (b) the noisy image artificially degraded with additive Gaussian noise ($\sigma = 20$) (c) isotropic smoothing (PSNR = 29.55 dB), (d) convolution (PSNR = 29.56 dB).

Figure 4 – Originally degraded image for denoising: (a) the noisy image, (b) isotropic smoothing and (c) convolution.

IV. CONCLUSION

In this study, we explained some solutions to PDEs and the Fourier transform for noise removal. These approaches reduce noises in grayscale images. However, image structures such as edges and textures are quickly blurred due to the fact that these are isotropic approaches. For this reason, other mathematical solutions using anisotropic diffusion PDEs [2]-[4] with fuzzy sets [5]-[6] will be examined in detail in the future.
REFERENCES


BIOGRAPHY

Belir Dizdaroglu received his BS, MS and PhD degrees from Department of Electrical and Electronics Engineering, Karadeniz Technical University (KTU), Trabzon, Turkey, in 1994, 1998 and 2007, respectively. He is currently working as an Assistant Professor in Department of Computer Engineering at KTU, Trabzon, Turkey.

His research interests include image inpainting, colorization and editing, fuzzy and wavelet-based image denoising, and old motion picture restoration.