RESEARCH ARTICLE | JUNE 07 2024

An anisotropic diffusion algorithm for image deblurring

Lorella Fatone ; Daniele Funaro

AIP Conf. Proc. 3094, 370004 (2024)
https://doi.org/10.1063/5.0210637

An Anisotropic Diffusion Algorithm for Image Deblurring

Lorella Fatone[1,a)] and Daniele Funaro[2,b)]

[1]*Dipartimento di Matematica, Università di Camerino, Via Madonna delle Carceri 9, Camerino, 62032, Italy.*
[2]*Dipartimento di Scienze Chimiche e Geologiche, Università di Modena e Reggio Emilia, Via Campi 103, Modena, 41125, Italy.*

[a)]Corresponding author: lorella.fatone@unicam.it
[b)]daniele.funaro@unimore.it

Abstract. This paper deals with the problem of image deblurring. A suitable discretization scheme for a particular nonlinear time-dependent partial differential equation of parabolic type is experimented. The method is implemented by reversing the arrow of time in order to damp diffusion. Only one step is enough to reconstruct the edges of a corrupted picture affected by average blur. Thus, the procedure turns out to be extremely efficient.

INTRODUCTION

Deblurring is a fundamental problem in image processing. Many successful methods rely on the use of Partial Differential Equations (PDEs). In this framework, images are represented in a continuous context and are analyzed through appropriate nonlinear time-dependent equations that are suitably discretized. Among many papers and books, we mention for instance [3], [7], [8], [1], [4], [6] and the references therein. A classical scheme is the one introduced by Perona & Malik in [5]. This is based on the following PDE:

$$\frac{\partial u}{\partial t} = \operatorname{div}\left(g_a(\|\nabla u\|)\, \nabla u \right), \tag{1}$$

with div u and ∇u denoting the divergence and gradient operator, respectively. The function g_a is defined, for a given parameter a, as:

$$g_a(\eta) = \left(1 + \frac{\eta^2}{a^2} \right)^{-1}. \tag{2}$$

In this way, the anisotropic diffusion coefficient in (1) depends locally on the magnitude of $\|\nabla u\|$, so that u is more diffused in the zones where there is less variation. Equation (1) is defined on a square and Neumann conditions are imposed on the boundaries. As far as the space variables are concerned, the approximation of (1) is usually approached by classical centered finite-differences based on a set of points (x_i, y_j) representing the image under study. For the time variable an explicit scheme of Euler type is commonly implemented. The algorithm is well-suited, for instance, in edge detection and segmentation problems.

As previously illustrated in [2], we follow here a similar approach. There are, however, a few significant changes with respect to the Perona & Malik technique. First of all, we propose a different PDE. Secondly, this new PDE is approximated in space by a high-order scheme constructed on two interlaced grids. Finally, the forward Euler scheme is applied by going backwards in time. For the reasons pointed out in [2], the resulting technique ends up to be rather efficient. We quickly describe the method in the next section, and we successively provide the results of some numerical tests.

International Conference of Numerical Analysis and Applied Mathematics
AIP Conf. Proc. 3094, 370004-1–370004-4; https://doi.org/10.1063/5.0210637

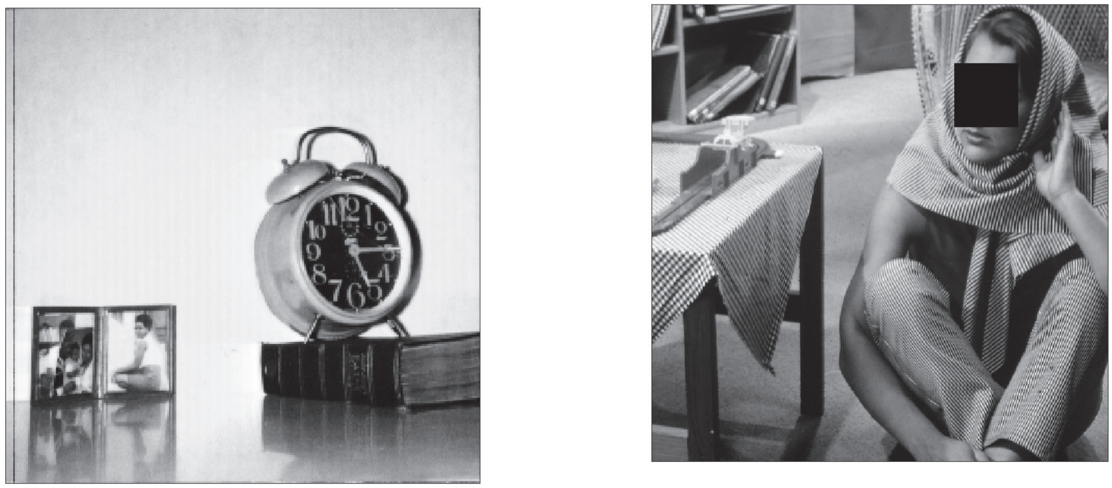

FIGURE 1. Images used for the tests

THE ALGORITHM

The new equation under study is the following one:

$$\frac{\partial u}{\partial t} = \sqrt{\frac{\|\nabla u\|^2}{1 + \|\nabla u\|^2}}\, \Delta u, \tag{3}$$

where Δu is the Laplace operator. Also in this case, u is defined on a square and Neumann boundary conditions are imposed.

Together with the standard grid points (x_i, y_j) we introduce intermediate $(x_{i\pm 1/2}, y_{j\pm 1/2})$ points, where the indices take semi-integer values. We refer to these new points as the "staggered grid". The algorithm is based on the following steps (the details are fully explained in [2]):

- Provide the original image values $u_{i,j}$ at the points (x_i, y_j);
- Build a centered discretization of order 3 of the space operator in (3) at the staggered-grid points. Denote the so obtained values by $R_{i\pm 1/2, j\pm 1/2}$;
- By polynomial interpolation of order 3, from the staggered to the classical grid, evaluate the quantities $R_{i,j}$;
- Update the image by setting $u_{i,j} = u_{i,j} + \gamma R_{i,j}$;
- Reiterate, if necessary.

The parameter γ should be negative. In this fashion, we are going backwards in time. Reversing time in nonlinear diffusion is, in general, an unstable process, at least for the continuous problem. Nevertheless, we suggest implementing the discrete scheme described above for just one or a few steps. In this way, there is no concern about stability issues. The parameter $\gamma < 0$ must be properly triggered. We do not have an exact recipe, but we suggest taking γ around 0.5% of the range of values attained by $u_{i,j}$. Note that in image reconstruction, the color palette for a grayscale image usually ranges from 0 to 255. More information is provided in [2].

NUMERICAL RESULTS

We present some numerical results concerning the deblurring of a couple of images. Figure 1 show the original pictures used for the tests. As a blurring procedure we adopt the one that averages the values in a suitable neighborhood of

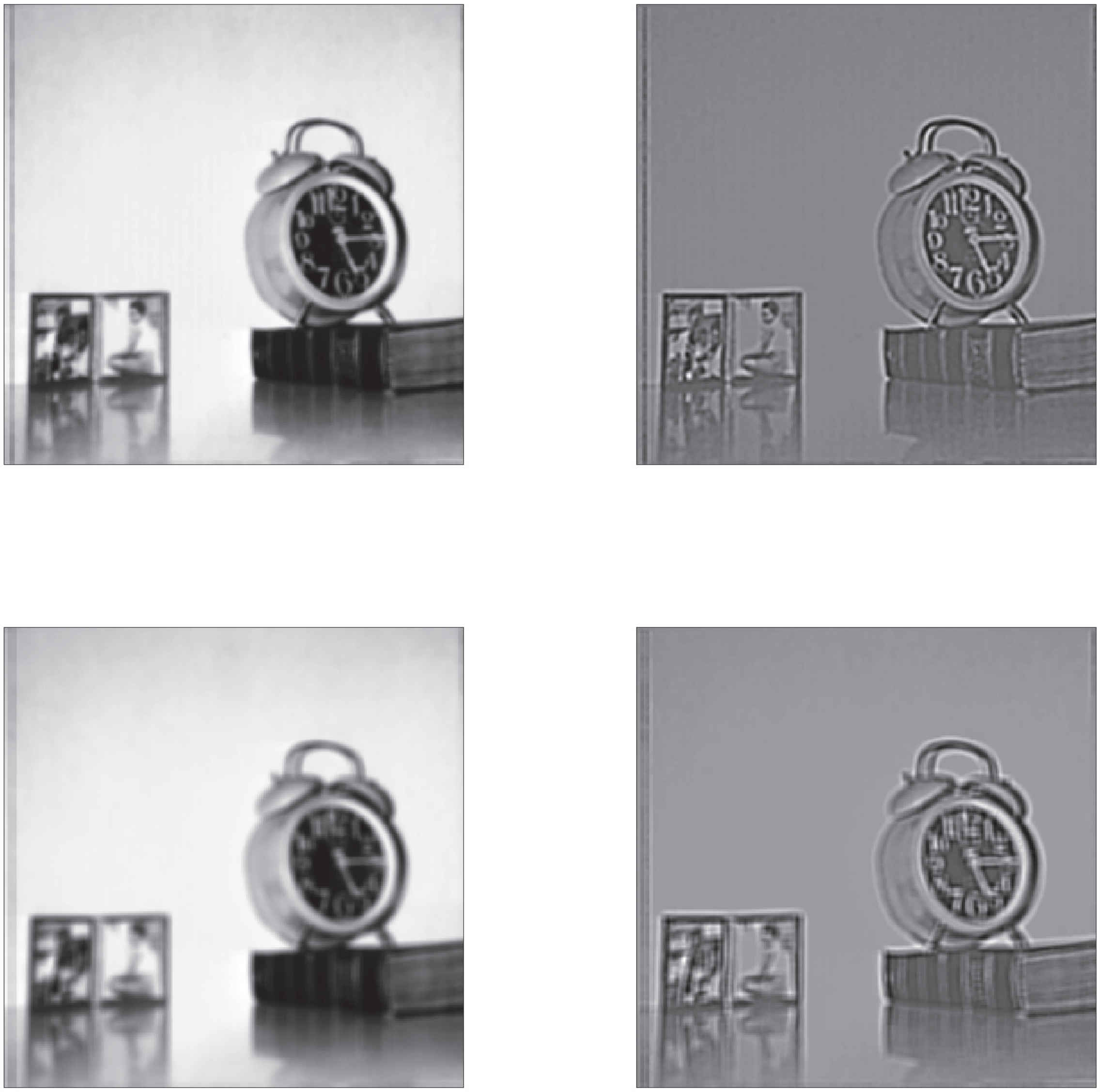

FIGURE 2. Blurred images have been generated with $\beta = 1$ (top-left) and $\beta = 2$ (bottom-left). The corresponding reconstructions obtained with the proposed algorithm are shown on the right. On top, the digits of the alarm-clock are clearly readable.

each pixel. In detail, the output pixel value in the blurred image is the arithmetic mean of the original pixel values belonging to a square of size $2\beta + 1$, $\beta \geq 1$, centered at each point. For example, if we choose $\beta = 1$ in the blurring procedure, then we average the value attributed to the pixel together with those of all the eight neighboring pixels. Note that we only use valid pixels when portions of the blurring matrix falls outside the original image.

Figures 2 and 3 show on the left some images polluted by different degrees of blurring, and on the right the refocused images obtained by applying a single step of the algorithm described in the previous section, with $\gamma = -10$.

The proposed procedure may be extended and adapted to take into account other various image deblurring techniques and more sophisticated known models for image denoising.

FIGURE 3. Blurred images have been generated with $\beta = 4$ (top-left) and $\beta = 5$ (bottom-left). The corresponding reconstructions obtained with the proposed algorithm are shown on the right. The details of the scarf and the tablecloth become visible.

REFERENCES

[1] S. Bonettini, F. Porta, V. Ruggiero and L. Zanni, J. Comput. Appl. Math **385**, 1-30 (2021).
[2] L. Fatone and D. Funaro, to appear in Ann. Univ. Ferrara (2022) (http://arxiv.org/abs/2204.13475).
[3] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 4th Edition (Pearson, New York, 2018).
[4] D. Lazzaro, E. Loli Piccolomini, V. Ruggiero and F. Zama, J Phys Conf Ser **904**, 012009-012015 (2017).
[5] P. Perona and J. Malik, IEEE Trans. Pattern Anal. Mach. Intell. **12**(7), 629-639 (1990).
[6] F. Porta, M. Prato and L. Zanni, J. Sci. Comput. **65**(3), 895-919 (2015).
[7] G. Sapiro, *Geometric Partial Differential Equations and Image Analysis* (Cambridge University Press, 2001).
[8] J. Weicker, *Anisotropic Diffusion in Image Processing*, ECMI Series (Teubner-Verlag, Stuttgart, Germany, 1998).