An evolution method for image restoration

Robert Acar and José Luis Calderón
University of Puerto Rico, Mayagüez

Abstract

A pre-processing step in image restoration consists of smoothing the image in order to reduce the noise. Regarding the grey-scale image as a mapping from a plane region to the unit interval (where the intensity of each pixel is a number ranging from zero to one), the evolution approach is to take the raw image as initial data, then solve, for small time, a certain partial differential equation. We analyse the properties of the Darboux operator to this effect; it is a hyperbolic operator, hence preserves discontinuities and edges, yet has intermediate smoothing properties which it shares with the traditional linear diffusion methods ("low-pass filters" in the Engineering literature).

1 Introduction

Image processing, the treatment of acquired images, has now become an extensive field of study. In the mathematical model, we assume that an image is a function over a certain domain, the frame, taking values in a certain range. In the case of grey-scale intensity image, the range is the interval $[0, 1]$. For stochastic models, replace single function by stochastic process (or "random field"). Good introductions to the field of mathematical image processing are [AK02], [CS05]. One of the tasks involved is image restoration, which typically consists of trying to diminish noise or blur imposed on the image. These two phenomena are usually distinct; blur is a systematic and repeatable process, imposed by the optical instrument, and typically modelled by convolution by a certain kernel. Noise is the effect of random natural phenomena.

One of the recent development in image denoising has been the use of PDE (partial differential equations) methods; we regard the acquired image

as the initial value for some initial-value problem associated with a partial differential equation, and we generate from it a time sequence of images, where time plays the role of an artificial parameter. For instance, solving the heat equation where the "dirty image" plays the role of initial data, is very effective at reducing the fluctuations due to random noise, but since the heat operator (convolution by the heat kernel) has isotropic effect, this not only reduces noise, but also blurs and erases the edges which make for the sharpness of the image. A fix around this is to use anisotropic diffusion; see [CS05], [AK02], [tHR94]. For an excellent survey of the scope and applicability of PDE methods, see also [LAM93].

A desirable feature then, of a PDE method, is to have some smoothing property. Before we continue, let us mention explicitly the models of noise which we use in our numerical experiments:

1. Gaussian noise is a field of pointwise random fluctuations, with Gaussian probability density function, each having zero mean, common variance and satisfying certain technical assumptions of isotropy of the spatial correlation of fluctuations. In this model, the noise is additive:

$$u(x) = \bar{u}(x) + n(x)$$

   where $\bar{u}$ is the ideal image and n is the noise.

2. Speckle noise uses the same model, but in a multiplicative way:

$$u(x) = \bar{u}(x)(1 + n(x))$$

3. Salt-and-pepper noise models the appearance of random grains on the television screens of time past, and consists of randomly coloring some dots on the image in black or white, according to a certain density:

$$u(x) = (1 - s(x))\bar{u}(x) + s(x)c(x)$$

   where both $s(x)$ and $c(x)$ are random variables taking the value 0 or 1.

This work was inspired by the article [Kic96], which lists certain results, but does not show the numerical experiments.

2 The Euler-Poisson-Darboux equation

The partial differential equation

$$u_{tt} + \frac{\alpha}{t} u_t - \Delta u = 0 \tag{1}$$

arises in solving the wave equation in higher dimensions (with $\alpha = n - 1$ if n is the dimension of space) and is referred to, in the literature, variously as the Euler-Poisson-Darboux equation ([DH53], [Wei55], [Joh55]) or the Darboux equation ([CH62]). We will name it thereafter the Darboux equation (keeping to the observation ascribed, maybe to Lagrange, that objects in mathematics are often named after the first person who rediscovers them after Euler. In this case, it would be the second). We will allow α to have values other than $n - 1$, but in the scope of this work, our numerical experiments use mostly $\alpha = n - 1 = 2$.

2.1 Equation over R^n

Over the entire space, the associated initial-value problem to (1) consists of specifying the value $f(x)$ of u at time zero, and the compatibility condition $u_t = 0$ at time zero (note that (1) is singular at the initial time). It is well-known (see [CH62]) that, for $\alpha = n - 1$, the solution to the initial-value problem consists of taking for $u(t, x)$ the spherical mean of f over the sphere of centre x and radius t. In [DH53], Diaz and Weinberger generalise this result to other values of α in two steps:

- For integer $\alpha > n-1$, they show, using the descent method of Hadamard, that the solution is still obtained by spherical means, or, more descriptively, cylindrical means over $\alpha - (n - 1)$ dimensional cylinders.

- By a continuation argument, they extend this formula to the whole complex plane, excluding negative odd value of α.

Two remarks:

1) Uniqueness hold for α with nonegative real part, but not otherwise.

2) By replacing t by $-t$, we see that the equation is reversible.

2.2 Equation over a rectangular domain

One surmises that, in the case of the initial-value problem over a bounded domain with periodic boundary conditions, one should derive similar results using elementary means. This turns out to be indeed the case. By imposing periodic boundary conditions over, say, the square $[0, 2\pi] \times [0, 2\pi]$, we look for solutions of the form

$$u(t,x) = \sum_{p \in Z^2} c_p(t) e^{ip\cdot x}, \quad \bar{c_p} = c_{-p}$$

where the condition on the coefficients expresses that u is real-valued. Transferred to the coefficients of the Fourier expansion, the Darboux equation becomes

$$tc''(t) + \alpha c'(t) + |p|^2 tc(t) = 0$$

where we suppressed the dependence of $c(t)$ on the mode p. The Frœbenius method then consists, for each p belonging to the integer lattice, of looking for c of the form

$$c(t) = \sum_{n=0}^{\infty} a_n t^{r+n}, \quad a_0 \neq 0.$$

Taking into account the initial-value conditions, and considering the resulting recurrence relations, we obtain that:

- For $\alpha > 0$, there is an unique solution (the case $\alpha = 0$ may be omitted, since it corresponds to the wave equation).

- For real, noninteger $\alpha < 0$, there are in general two linearly independent solutions. If α is a negative integer, then there are infinitely many solutions if α is even, and, in general, no solutions if α is odd.

- A result mentioned in [Kic96] is that, if $f(x) \in H^s(R^2)$, then, for each positive time, $u(t,x) \in H^{s+1/2}(R^2)$. In the context of the periodic boundary-value problem, one obtains this result by observing that (1) if of Bessel type, and by using the characterisation of $H^s_p(R^2)$ (space of periodic H^s functions) via the Fourier transform: if $v(x) = \sum_{p\in Z^2} c_p e^{ip\cdot x}$, then

  $$v \in H^s(R^2) \Leftrightarrow \sum_{p \in Z^2} (|p|^2)^{s/2} c_p \in l^2$$

  and the fact that $J_0(t|\xi|)$ decays at infinity like $|\xi|^{-1/2}$.

It is this intermediate smoothing property which lends this method its interest. Of course, we must keep in mind that the scale of spaces H^s, which allows easy analysis of smoothing, is not the natural one where to consider images, along with their edges and discontinuity types. In this scale, the adequate space where to chose f would be H^0. On the other hand, studying the smoothing properties of the Darboux operator in the space BV is beyond the scope of the present work.

3 Numerical results

For a more complete account of the numerical experiments, we refer to [Cal09]. In this summary, we merely highlight some of the conclusions, and state some additional remarks.

The results we show use two images, one synthetic, containing high-level information, the other, cropped from a photograph.

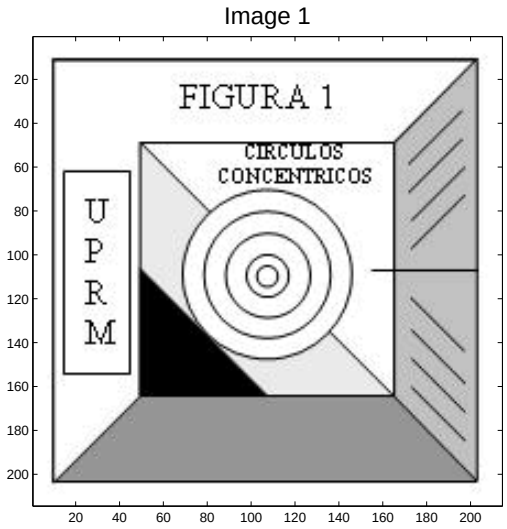

Figure 1:

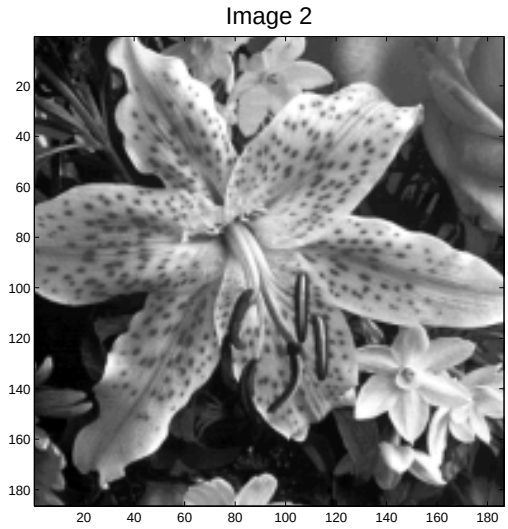

Figure 2:

3.1 Discrete scheme

We used an explicit finite-difference method, choosing uniform space and time steps $\Delta x = \Delta y$ and Δt. The first time derivative is approximated by a

single-step formula

$$u_t^n = \frac{u^n - u^{n-1}}{\Delta t}$$

which is of order of accuracy $O(\Delta t)$, and the second time derivative by

$$u_{tt}^n = \frac{u^n - 2u^{n-1} + u^{n-2}}{\Delta t^2},$$

and the Laplacian of u, by the standard four-point discrete Laplacian. Solving the periodic boundary-value problem makes it easy to deal with the boundary. An analysis of stability indicates that the scheme is stable if $\lambda < 0.5$, λ being the ratio $\Delta t/\Delta x$. One drawback of this simple scheme is that it does not strictly preserve the range of the image: the weights involved in computing u^n from u^{n-1} and u^{n-2} are affine, but not convex, as some are negative. Since we only use the method for small time, the effect is not significant.

Of concern is the effect of the choice of time step over the value of u at a certain time. Or, put another way, while the ideal image is over the unit square, the actual instance of the image is as a matrix of pixels, the size of which is determined by the resolution. The value of $u(t)$ at time t should depend essentially on t, but not on the number of time-steps required to reach t. The following tables list, as a first column, the amount of noise (at time zero) in the L^1 norm, then, the discrepancy between u and $\tilde{u}$ at a chosen fixed time, with u corresponding to $\lambda = 0.4$ and $\tilde{u}$ corresponding to $\lambda = 0.1$. The difference in u due to different values of λ is seen, at least, not to exceed the size of the noise. In the remaining experiments, we chose $\lambda = 0.2$.

Table 1: Image 1

	noise	$\lvert u - \tilde{u}\rvert_1$	$\lvert u - \tilde{u}\rvert_\infty$
speckle	0.09	0.02	
gaussian	0.05	0.01	
salt & pepper	0.03	0.02	
no noise		0.01	0.14

Table 2: Image 2

	noise	$\|u-\tilde{u}\|_1$	$\|u-\tilde{u}\|_\infty$
speckle	0.08	0.01	
gaussian	0.08	0.01	
salt & pepper	0.02	0.01	
no noise		0.007	0.07

3.2 Quantifying denoising

A measure used in signal processing, for signal denoising or compression, is the "signal-to-noise ratio", and the closely related "peak signal-to-noise ratio". The aforementioned are usually defined using the squared-error, but it is possible to define a related measure relying on the L^1 error. We choose the following:

$$SNR = -100 \log(|u-\bar{u}|_1),$$

where u is the restored image, and $\bar{u}$ the clean image. Of course, in practice this quantity is unknown (for lack of knowing $\bar{u}$), but it is still of interest to check whether the empirical measure in question agrees with subjective perception in the case of images. For certain iterations of the Darboux process, we list the corresponding SNR. In order better to compare, we used noise density and variance respectively of 0.08 and 0.02 for speckle and Gaussian noise. For image 1, we obtain:

Table 3: Image 1, speckle noise

iteration	2	3	4	5	6	7
SNR	211.32	212.28	212.97	213.18	212.77	211.65

Table 4: Image 1, gaussian noise

iteration	2	3	4	5	6	7
SNR	259.80	260.37	260.20	259.06	256.79	253.37

We then show side-by-side the images corresponding to highest SNR, and best perceived fit (figures 3, 4, 5, 6).

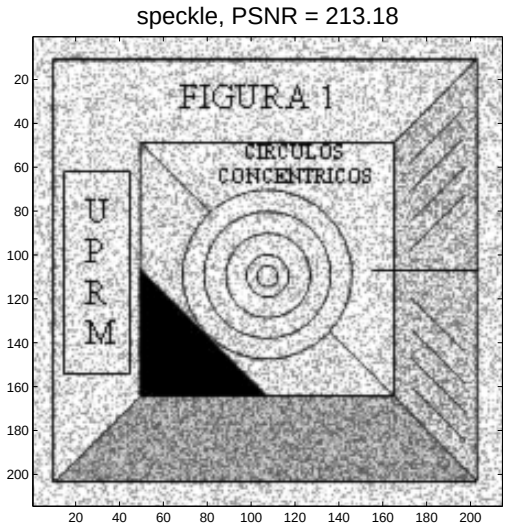

Figure 3:

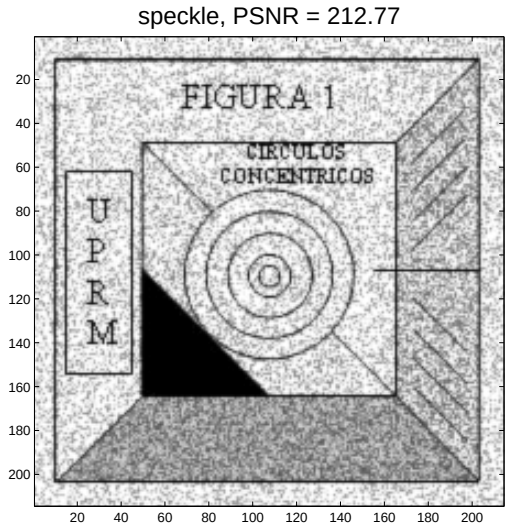

Figure 4:

For image 2, we used speckle noise only. In the case of this image with lower high-level information content, it is our impression that, while the SNR seems roughly to correlate with perception, the agreement is by no means exact (figures 7, 8, 9, 10).

Table 5: Image 2, speckle noise

iteration	4	8	9	10	11	12	13
SNR	242.56	283.54	290.13	292.12	289.96	285.05	279.01

3.3 Comparison with filters

A traditional denoising method in engineering involves the use of filters. Filtering an image consists in applying to it a local averaging-like operator, defined by a fixed mask of a certain size. The smaller the size, the more local the operator. If the effect of the filter is true averaging, then this amount to convolution by a certain kernel. It is a linear operation, and generalises in a certain sense solving the heat equation. The effect of such a filter is to diminish the noise (by decreasing random oscillations) but also to blur separation between image regions, akin to what isotropic diffusion does. A

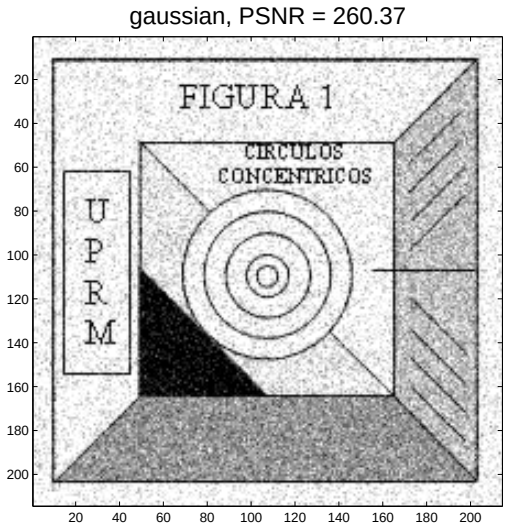

Figure 5:

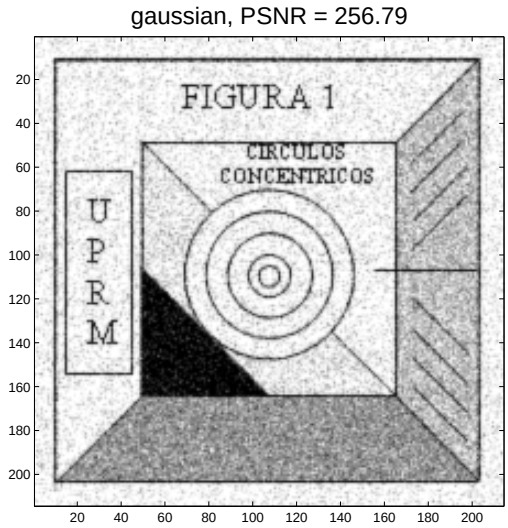

Figure 6:

fix around this is to use a "median filter", which is nonlinear, and uses order statistics instead of convolution. It has the advantage of preserving boundaries between solid segments (i.e., regions of nonzero measure), but not thin curves. As a contrast to the effect of evolving by the Darboux process, we show the effect of using filters to both images. For both filters, we chose masks of size 3×3, having the smallest support; the choice of larger masks would further aggravate blurring. It must be mentioned that another difference between a PDE based method such as this, and filter methods, is that while the filter is applied to get a single restored image, and is an irreversible procedure, the evolution method generates a sequence of smoothed images along the time scale. See figures 11, 12, 13, 14, 15, 16, 17 and 18.

3.4 Other choices of parameter

As mentioned earlier, negative values of the parameter α yield multiple solutions of the Darboux equation. This is not necessarily the case for the discretisation, but it is of interest to compare the effect. We show a sample result for $\alpha = -0.2$, starting from a clean image: figures 19, 20.

As can be inferred from the analysis in [DH53], the higher the value of α, the more smoothing the effect of the solution operator. Extensive tests for

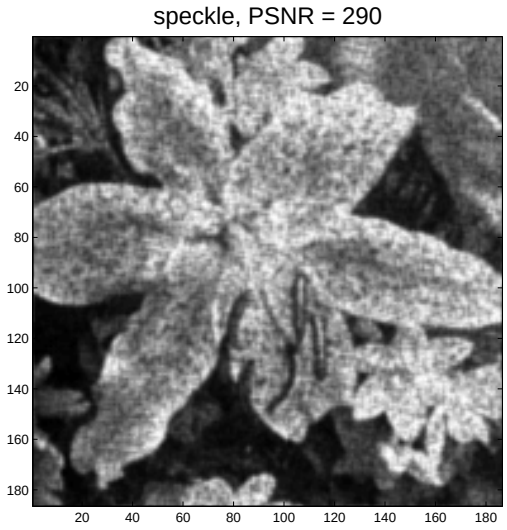

Figure 7: 9 iterations

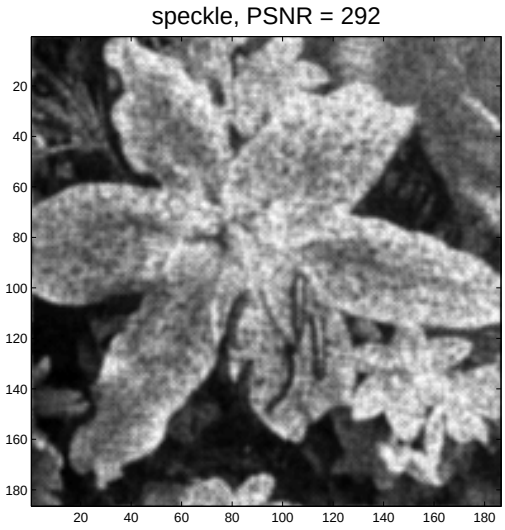

Figure 8: 10 iterations

different values of α are beyond the scope of this work, but we show below some sample results, also starting from a clean image. See figures 21, 22, 23, 24.

The reversibility property is of potential use. Because of the singularity of the Darboux equation at time zero, we cannot expect the discretised method to revert all the way back to the initial time. We show the result of running a certain number of iterations forward, then backtracking: figures 25, 26

However, by increasing the value of λ, we obtain a striking effect, of potential use in edge enhancement: see figure 27.

Acknowledgement We thank Prof. K. Rózga for pointing out the result about decay of the Bessel function of order zero, and its role in proving the smoothing property.

References

[AK02] G. Aubert and P. Kornprobst. *Mathematical problems in image processing*. Springer-Verlag, 2002.

[Cal09] J.L. Calderón. Hyperbolic smoothing for digital image restoration (in Spanish). Master's thesis, University of Puerto Rico, 2009.

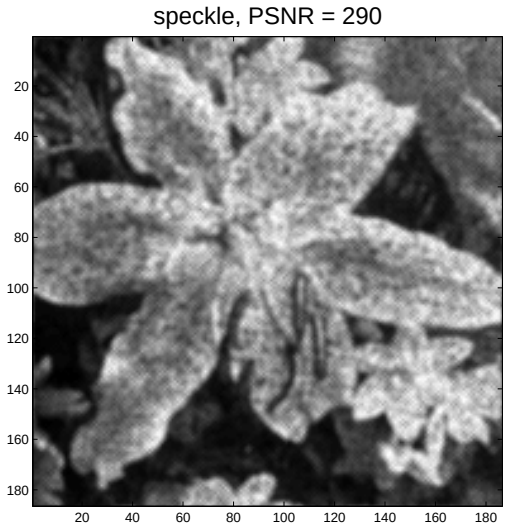

Figure 9: 11 iterations

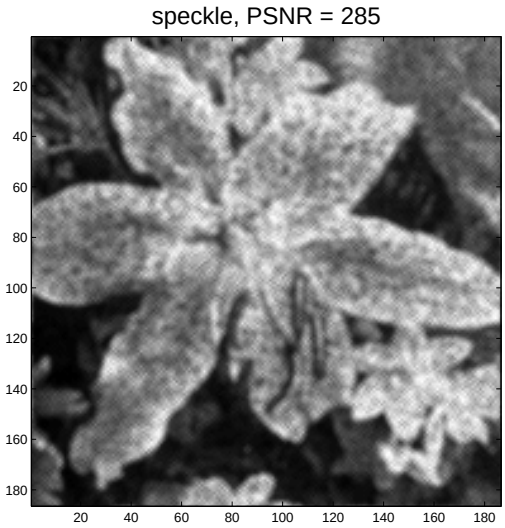

Figure 10: 12 iterations

[CH62] R. Courant and D. Hilbert. *Methods of mathematical physics*, volume II. Interscience, 1962.

[CS05] T.F. Chan and J. Shen. *Image processing and analysis.* SIAM, 2005.

[DH53] J.B. Diaz and H.F.Weinberger. A solution of the singular initial value problem for the Euler-Poisson-Darboux equation. *Bulletin of the AMS*, 1953.

[Joh55] F. John. *Plane waves and spherical means applied to partial differential equations.* Interscience, 1955.

[Kic96] S. Kichenassamy. Nonlinear diffusions and hyperbolic smoothing for edge enhancement. In *ICAOS '96*, number 219 in LNCIS, pages 119–124. Springer-Verlag, 1996.

[LAM93] P.L. Lions L. Alvarez, F. Guichard and J.M. Morel. Axioms and fundamental equations of image processing. *Arch. Rational Mechanics*, 123:199–257, 1993.

[tHR94] B.M. ter Haar Romeny, editor. *Geometry-driven diffusion in computer vision.* Kluwer, 1994.

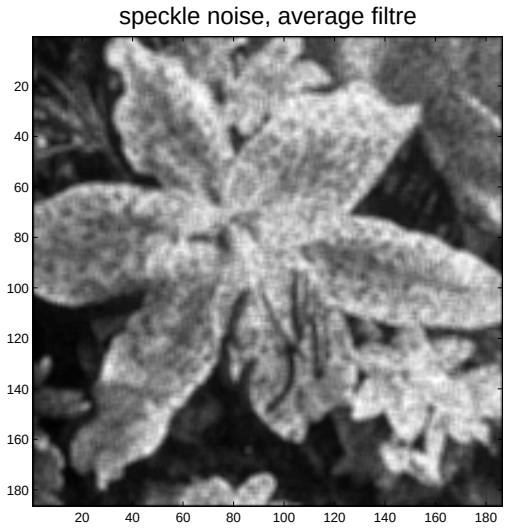

Figure 11:

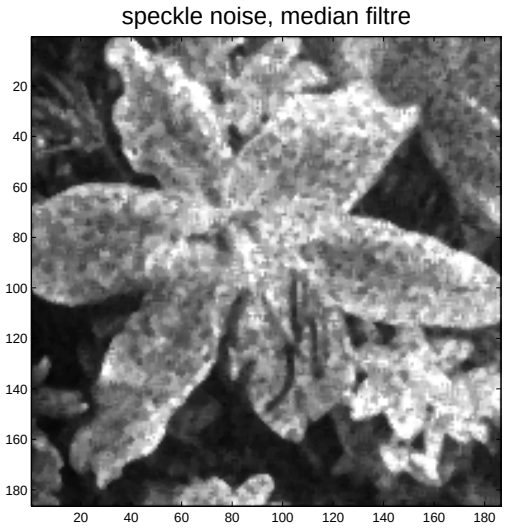

Figure 12:

[Wei55] A. Weinstein. The generalized radiation problem and the Euler-Poisson-Darboux equation. *Summa Brasiliensis*, 3, 1955.

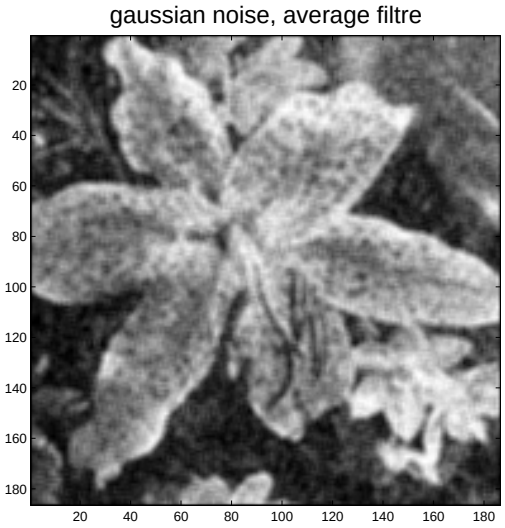

Figure 13:

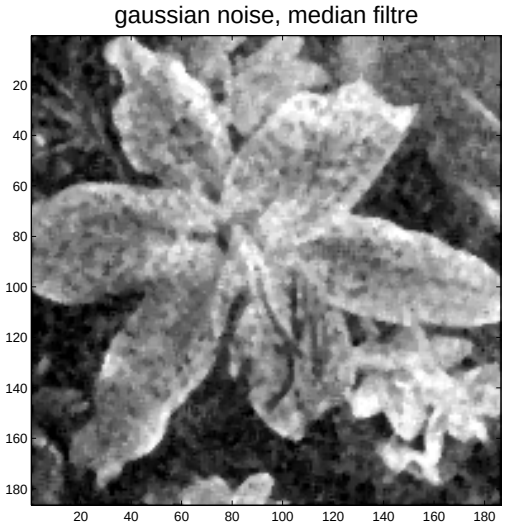

Figure 14:

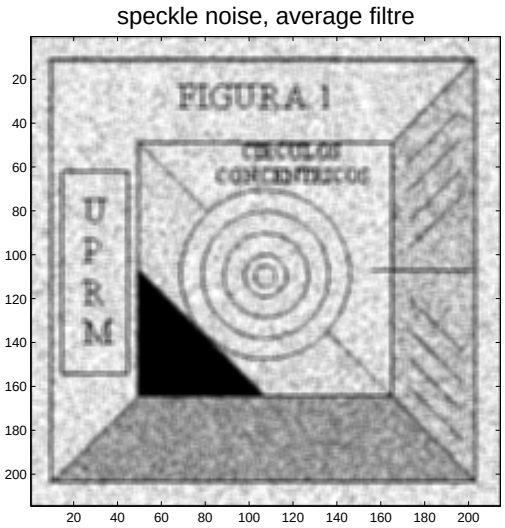

Figure 15:

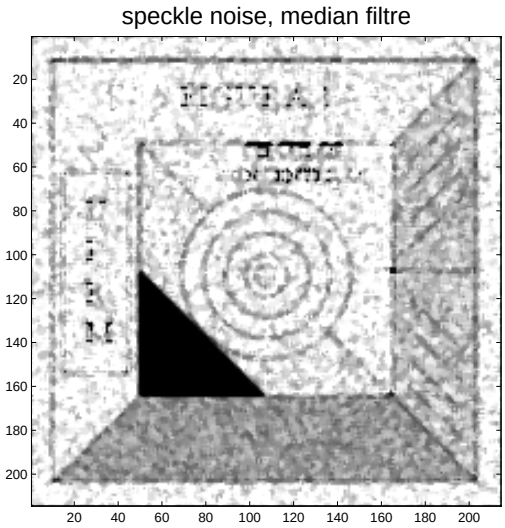

Figure 16:

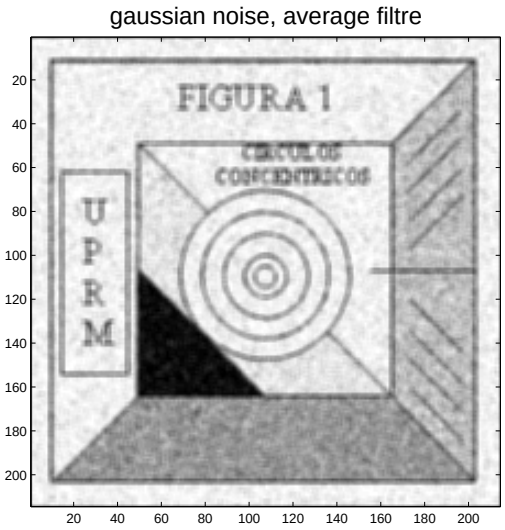

Figure 17:

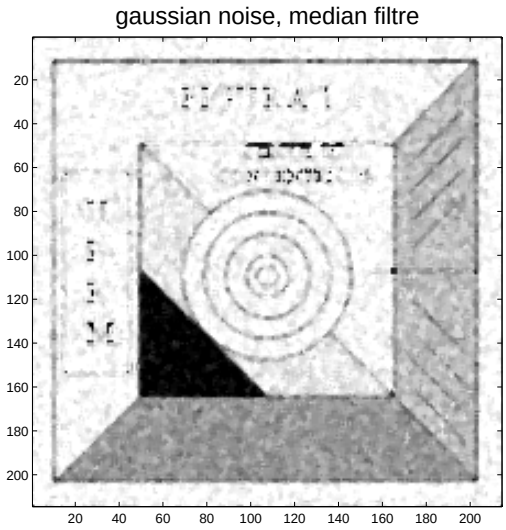

Figure 18:

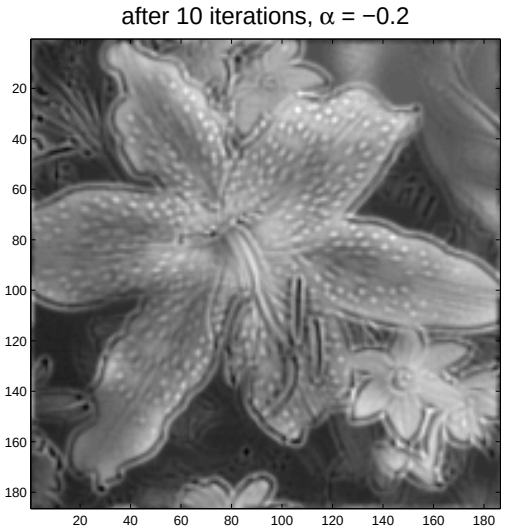

Figure 19:

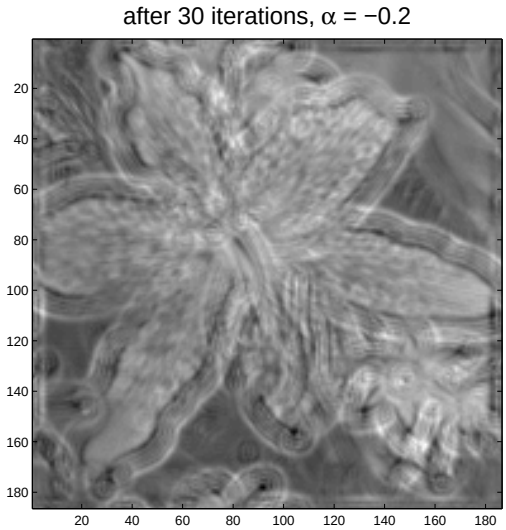

Figure 20:

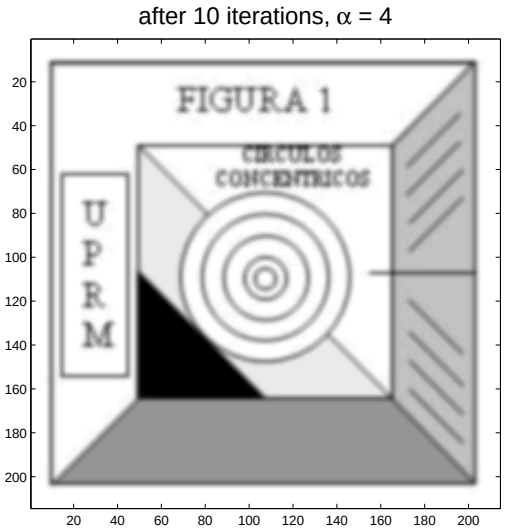

Figure 21:

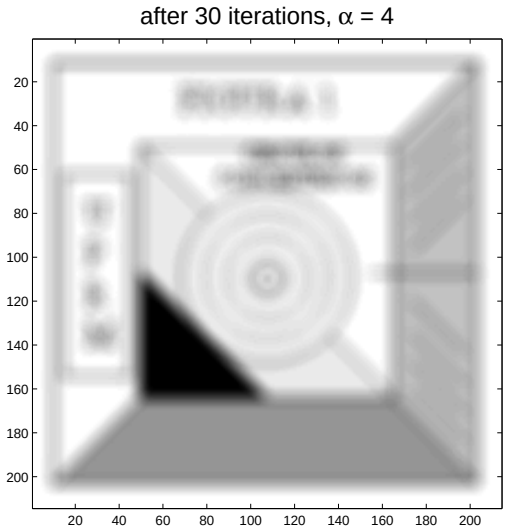

Figure 22:

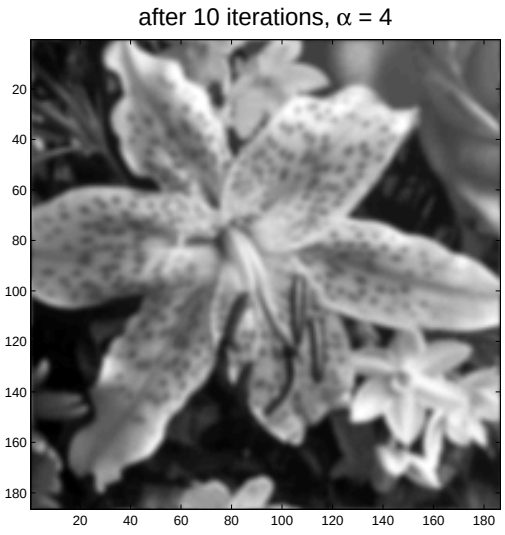

Figure 23:

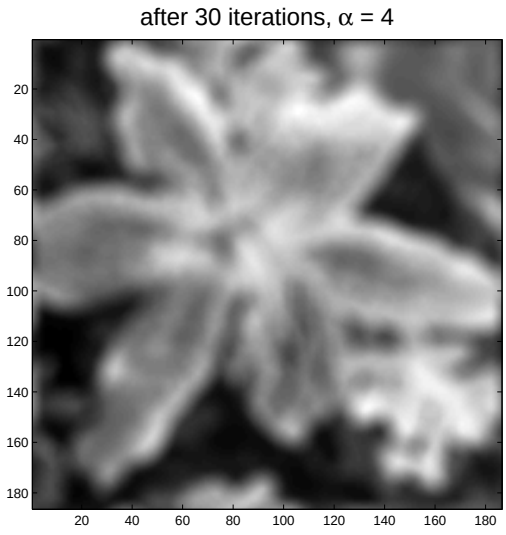

Figure 24:

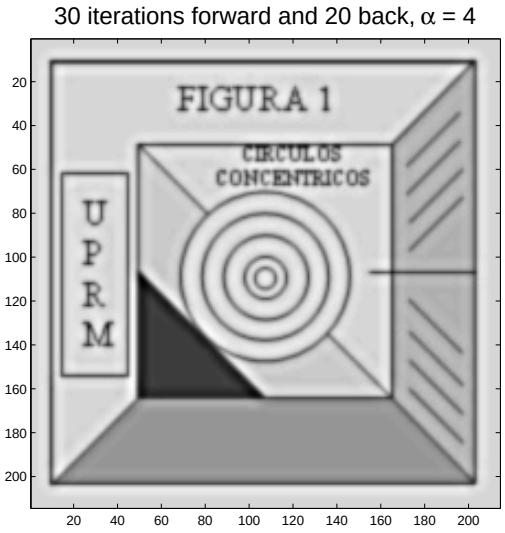

Figure 25:

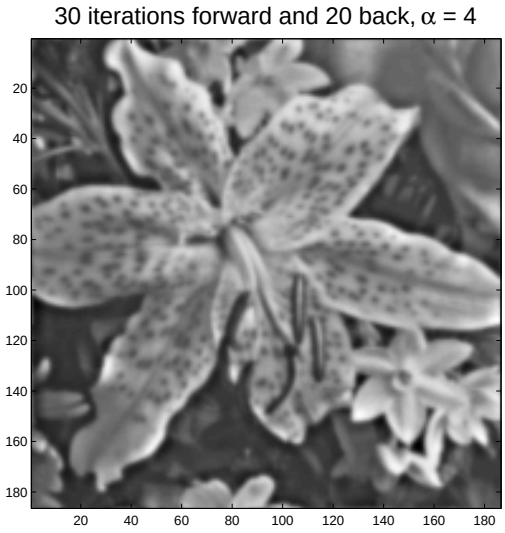

Figure 26:


15 iterations forward and 10 back, alpha = 4
180
160
140
120
100
80
60
40
20
20
40
60
80
100
120
140
160
180

Figure 27: