Wavelet despiking of fractographs

Jean-Marie Aubry and Naoki Saito

University of California at Davis,
Davis, California, USA

ABSTRACT

Fractographs are elevation maps of the fracture zone of some broken material. The technique employed to create these maps (confocal laser microscopy) often introduces noise composed of positive or negative 'spikes' that must be removed before further analysis. Since the roughness of these maps contains useful information, it must be preserved. Consequently, conventional denoising techniques cannot be employed. We use continuous and discrete wavelet transforms of these images, and the properties of wavelet coefficients related to pointwise Hölder regularity, to detect and remove the spikes.

Keywords: Fractography, despiking, Hölder exponents

1. INTRODUCTION

1.1. Fractographs

The study of material fractures is a subject of huge importance for the industry. In the particular case that we studied, jet engine manufacturers are concerned with the fatigue fractures occuring on turbine blades, with potential catastrophic effects. Fatigue occurs when the material (titanium, in this case) is subject to repeated stress variations due to cyclic constraints. The cost of these fractures, in terms of aircraft readiness and human lives, is pushing companies and governmental agencies to investigate this problem in detail.

One approach to this problem, known as *fractography*, is a diagnostic method aiming at deducing the conditions (load history) that propagated the fracture, from detailed images of the fracture zone. These images can be of different nature: for instance SEM (Scanning Electron Microscope) images, or elevation maps obtained by Confocal Laser Microscopy.

1.2. CLM elevation mapping

Elevation mapping of fracture surfaces through a confocal laser microscope is a technique developed by SRI International.[1,2] A laser beam is focused through the microscope at some point above the studied surface, which is raised (z direction) until the reflected light attains a maximum of intensity (the value of the reflected intensity can be kept as a by-product information). This gives the altitude of one point of the surface; the full mapping is obtained using a scanning in the x_1 and x_2 directions. This technique produces very precise elevation maps such as Figure 1, where the grey levels represent the altitude of the fracture surface. The resolution here is 0.233 μm/pixel in both x_1 and x_2 directions and 0.293 μm/grey level in the z direction.

However, it can happen that, if for instance the reflected light is scattered by a very high surface gradient, no clear maximum for the reflected intensity is found. In this case, the acquisition device retains an unspecified (random) value. On the resulting image, this creates white or black dots, as in the regions indicated by arrows on Figure 1; if the data is viewed in three dimensions, these points appear as protruding 'spikes'.

Note that this problem also occurs in several other applications, such as hyperspectral satellite images or interferometric fractography, where images are obtained by scanning some sensor returning unspecified numbers when 'missing values' happen.

E-mail addresses: jihem@ucdavis.edu (J.-M. A.), saito@math.ucdavis.edu (N. S.)
Web pages: http://math.ucdavis.edu/~

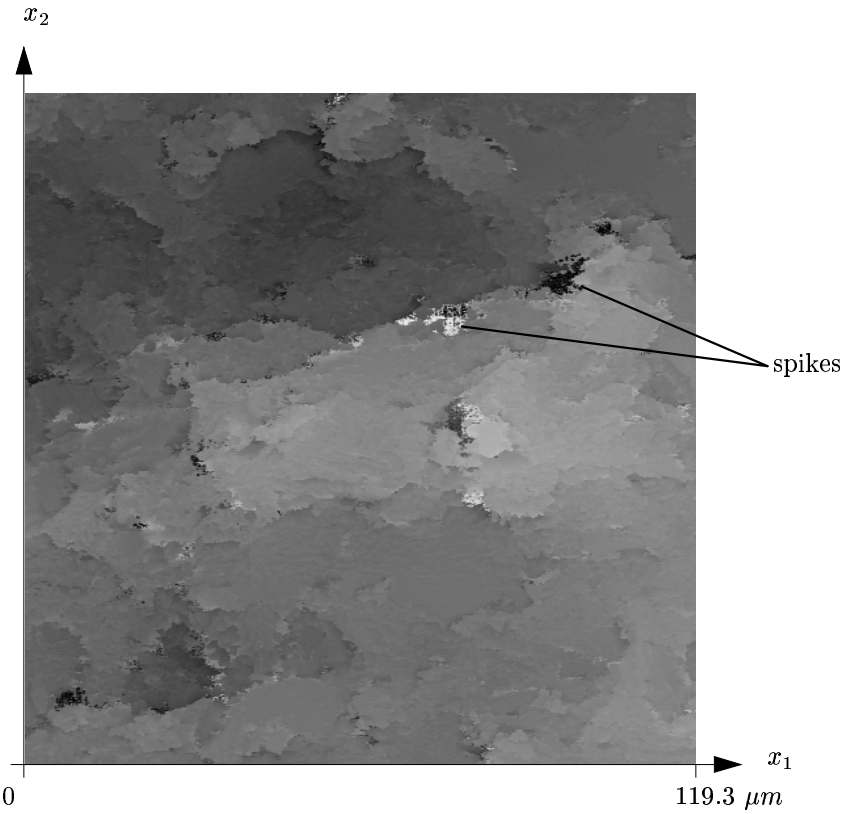

Figure 1. Elevation map obtained by Confocal Laser Microscopy

1.3. Surface roughness and ΔK estimation

These elevation maps are used to estimate an important fatigue parameter noted ΔK, which is the difference between minimum and maximum intensities of the stress applied to that region during the fatigue process. This ΔK parameter may be related to the surface roughness, measured for instance by spectral or wavelet analysis.[3] Since the surface roughness is the important data here, it is clear that the presence of artificial spikes is an obstacle to precise measurement; it is therefore necessary to eliminate them in a pre-treatment phase.

Note also that conventional denoising techniques, such as convolution with a regular kernel or wavelet coefficients thresholding, which tend to smooth out the details of the image, are not applicable here, precisely because the roughness contains the useful information. We thus decided to adopt a 'detect and repair' strategy, using the properties of the wavelet transform.

2. SPIKE DETECTION

2.1. Hölder exponents

The first task to achieve is to detect the points where the image is damaged by spikes. Spikes, corresponding to isolated values of a function, are points where this function is very irregular. Pointwise regularity can be measured by the pointwise Hölder exponent, defined as follows.

DEFINITION 1. *A function $f \in L^\infty$ is in the Hölder space $C^h(x_0)$ if there exists a polynomial p such that, for x in a neighborhood of x_0,*

$$|f(x) - p(x)| = \mathcal{O}(|x - x_0|)$$

The Hölder exponent of f at the point x_0 is

$$h_f(x_0) := \sup \{h : f \in C^h(x_0)\}$$

This definition is valid in two dimensions with $x := (x_1, x_2)$.

2.2. Continuous wavelet transform

Hölder exponents can be computed using discrete or continuous wavelet transform.[4] However, numerically, the latter gives better results since it allows to follow lines of local maxima for the modulus of wavelet coefficients.[5]

Let $\psi : \mathbb{R}^2 \to \mathbb{R}$ be a function with zero integral. We define the L^1-normalized scaled and translated wavelets

$$\psi_{a,b}(x) := a^{-2}\psi\left(\frac{x-b}{a}\right)$$

for $a > 0$ and $b \in \mathbb{R}^2$. The wavelet coefficients of f are $C_{a,b} := \langle f, \psi_{a,b}\rangle$.

DEFINITION 2. *We say that f is* weak uniform Hölder *if for all $n > 0$, there exist C_n such that for all $0 < a < 1$,*

$$|C_{a,b}| \leq -\frac{C_n}{(\log(a))^n}.$$

If f is weak uniform Hölder*, its Hölder exponents are related to the modulus of its wavelet coefficients by

$$h_f(x_0) = \liminf_{a\to 0} \inf_b \frac{\log(|C_{a,b}|)}{\log(a + |b - x_0|)} \tag{1}$$

provided that ψ has at least $\lfloor h_f(x_0)\rfloor + 1$ vanishing moments and continuous derivatives, each of them having fast decay.

2.3. Numerics

In this section we give some details about the numerical computation of continuous wavelet coefficients.

2.3.1. Discretization

Suppose the data f is given as a $2^J \times 2^J$ matrix ($J = 9$ on Figure 1). To do the transform, the scaled wavelet itself must be discretized into a matrix W_a. However, simple sampling ($W_a(x) = \psi_{a,0}(x)$) does not preserve — especially at small scale — the most important feature of the wavelet, namely its zero integral. To correct this, we take

$$W_a(x) = \int_{x_1-1/2}^{x_1+1/2}\int_{x_2-1/2}^{x_2+1/2} \psi_{a,0}(t)dt_2dt_1$$

so that $\sum_x W_a(x) = 0$. In the case of the popular 'Mexican hat' (negative Laplacian of the Gaussian) choice for ψ, these values can even be computed analytically:

$$\begin{aligned} W_a(x) :=& \frac{\sqrt{\pi}}{2a}\left(\operatorname{erf}\left(\frac{x_1+1/2}{a}\right) - \operatorname{erf}\left(\frac{x_1-1/2}{a}\right)\right) \\ & \left(e^{-\left(\frac{x_2+1/2}{a}\right)^2} + e^{-\left(\frac{x_2-1/2}{a}\right)^2} + 2x_2\left(e^{-\left(\frac{x_2+1/2}{a}\right)^2} - e^{-\left(\frac{x_2-1/2}{a}\right)^2}\right)\right) \\ & + \frac{\sqrt{\pi}}{2a}\left(\operatorname{erf}\left(\frac{x_2+1/2}{a}\right) - \operatorname{erf}\left(\frac{x_2-1/2}{a}\right)\right) \\ & \left(e^{-\left(\frac{x_1+1/2}{a}\right)^2} + e^{-\left(\frac{x_1-1/2}{a}\right)^2} + 2x_1\left(e^{-\left(\frac{x_1+1/2}{a}\right)^2} - e^{-\left(\frac{x_1-1/2}{a}\right)^2}\right)\right) \end{aligned}$$

where $\operatorname{erf}(u) = \frac{2}{\sqrt{\pi}}\int_0^u e^{-v^2}dv$ is the standard error function.

In this setting, we also can determine what the smallest acceptable scale is (a crucial question in all numerical applications of wavelets). If the scale a is too small, then the 'essential support' of our wavelet will be in the square

*This condition is also necessary, as shown in Ref. 6.

$[-\frac{1}{2}, +\frac{1}{2}] \times [-\frac{1}{2}, +\frac{1}{2}]$, hence, with our discretization choice, W_a will be everywhere (almost) zero, which is clearly an artifact. This effect starts appearing when the negative part of the wavelet enters this elementary square, and we should not take any smaller scale. The smallest scale $a_{\min}$ is thus the one which maximizes $W_a(0,0)$; for our choice of ψ, we computed

$$a_{\min} = 0.5632415942907933.$$

2.3.2. Periodization

Edges effects are a common problem with the wavelet transform. The simplest setting assumes that the function to be transformed is periodic. If this is not the case, artificial discontinuities are introduced on the edges, which would balefully interfere with detecting the spikes. In order to avoid this, we first apply a *smooth periodization* procedure using periodized folding and unfolding operators in both x_1 and x_2 directions.[7,8]

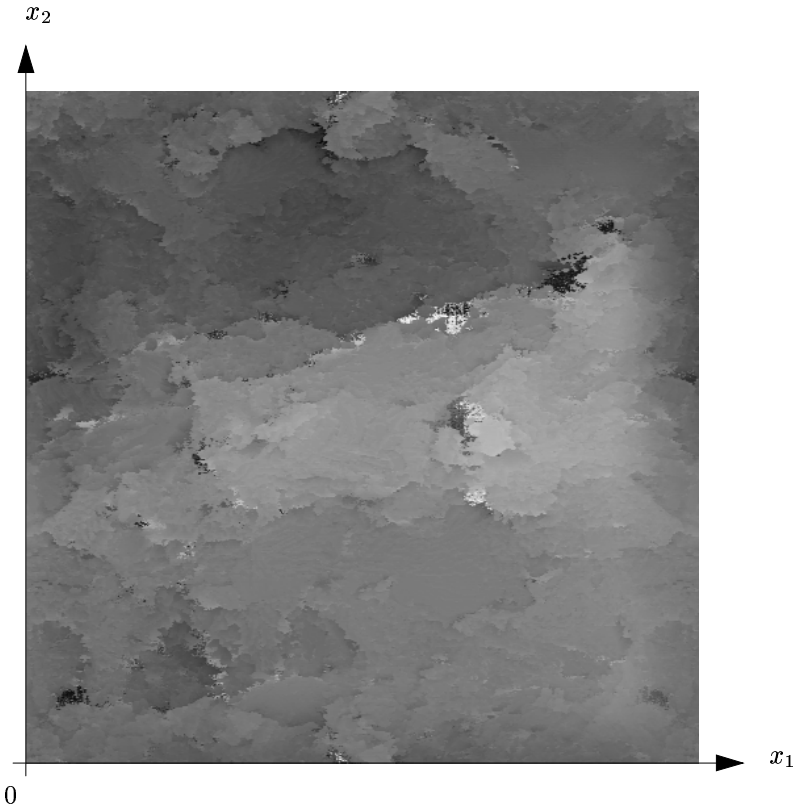

Figure 2. Smoothly periodized fractograph

This transformation is unitary, thus easily invertible, and allows us to recover the original image (minus the spikes) after treatment.

If f is $2^J \times 2^J$ periodic, $C_{a,b} = \int_{-2^J}^{2^J} \int_{-2^J}^{2^J} f(x)\tilde{\psi}_{a,b}(x)$, where $\tilde{\psi}_{a,b}$ is the periodized wavelet

$$\tilde{\psi}_{a,b}(x) := \sum_{k \in \mathbb{Z}^2} \psi_{a,b}(x - 2^J k)$$

Thanks to the fast decay of the wavelet chosen here, we get a reasonable approximation (at least for scales $a \leq 2^{J-2}$) of this infinite sum by taking only 4 values for k:

$$\tilde{W}_a(x) := W_a(x) + W_a(x - (2^J, 0)) + W_a(x - (2^J, 2^J)) + W_a(x - (0, 2^J))$$

2.3.3. Convolution

At a given scale a, the matrix $C_a(b) = C_{a,b}$ is simply the convolution $f * \tilde{W}_a$. Instead of doing this $\mathcal{O}(n^2)$ operation ($n = 2^{2J}$ being the number of evaluation points) directly, we do a fast convolution employing the (two-dimensional) Fast Fourier Transform:

$$C_a = \mathrm{FFT}^{-1}\left(\mathrm{FFT}(f) \times \mathrm{FFT}(\tilde{W}_a)\right)$$

which is an $\mathcal{O}(n \log(n))$ computation.

2.4. Detection algorithm

2.4.1. Wavelet coefficients thresholding

As a first approximation, (1) indicates that singularities of the function will correspond to the zones of large wavelet coefficients at small scale. We can use this remark to save computation time, by first thresholding the wavelet coefficients at the smallest scale, keeping only 5% (for instance) of the locations where the wavelet coefficient modulus are the largest. Not all of those points are spikes, however, and finer distinction will be made later using the size of wavelet coefficients at other scales, by computing the Hölder exponent. However, we are quite confident that all the spike locations were captured.

2.4.2. Computation of pointwise Hölder exponents

The second step consists in computing the pointwise Hölder exponent at each of the previously selected locations. As previously mentioned, this can be done by following lines of local maxima of the wavelet coefficients. But, even though the selected wavelet coefficients at the smallest scale are among the top 5%, they are not necessarily local maxima, and thus not connected to a line of local maxima. To work around this difficulty, we make another hypothesis, which is that the large wavelet coefficients generated by a singularity at x_0 stay in its *cone of influence*, that is the set of b such that $|b - x_0| \leq a$. Note that this is not true for all singularities; exceptions to this hypothesis are *chirps* or *oscillating singularities*.[9]

With this hypothesis, we can take the largest coefficient in the cone of singularities at increasing scales until we catch a line of local maxima (that is, until the largest coefficient is found in the interior of the disk $D(x_0, a)$).

For each point x_0, we thus constructed a line of wavelet coefficients $C_{a,b(a)}, a > a_{\min}$ on which the pointwise Hölder exponent can be computed using (1).

REMARK 1. *Finding the maximum wavelet coefficient modulus in the cone of influence of each singularity is costly: at each scale a, we have to look in a disk of radius a, which is an $\mathcal{O}(a^2)$ operation. But in general only the first few scales are needed until a line of local maximum is reached.*

2.4.3. Hölder exponent thresholding

Spikes correspond theoretically to points of Hölder exponent $h(x_0) = 0$. In practice, due to the finite-scale effects, a positive Hölder exponent is observed at spike locations. We chose a threshold below which the Hölder exponent indicates a spike: empirically, $h = 0.2$ gives good results (Figure 3).

3. SPIKE REMOVAL AND IMAGE RECONSTRUCTION

Once spikes are detected, the fractograph needs to be 'repaired' at those points. For the present application, the roughness of the surface is the important parameter that needs to be preserved, so we chose a procedure to reconstruct the missing values such that Hölder exponents are of the same order as in the neighborhood. Visually, this amounts to filling in the missing regions with the surrounding texture (Figure 4).

Note that this section is completely independent from the previous one: we simply assume that the damaged pixels were detected on the image, and the locations are stored in a matrix D ($D(x) = 1$ if the pixel is damaged, $D(x) = 0$ else). It is not unreasonable to hope that this first step might be somehow incorporated in the elevation map acquisition process, having a NaN (Not a Number) value returned instead of a random number, which would of course make Section 2 obsolete. Nevertheless, the repairing action would still be needed.

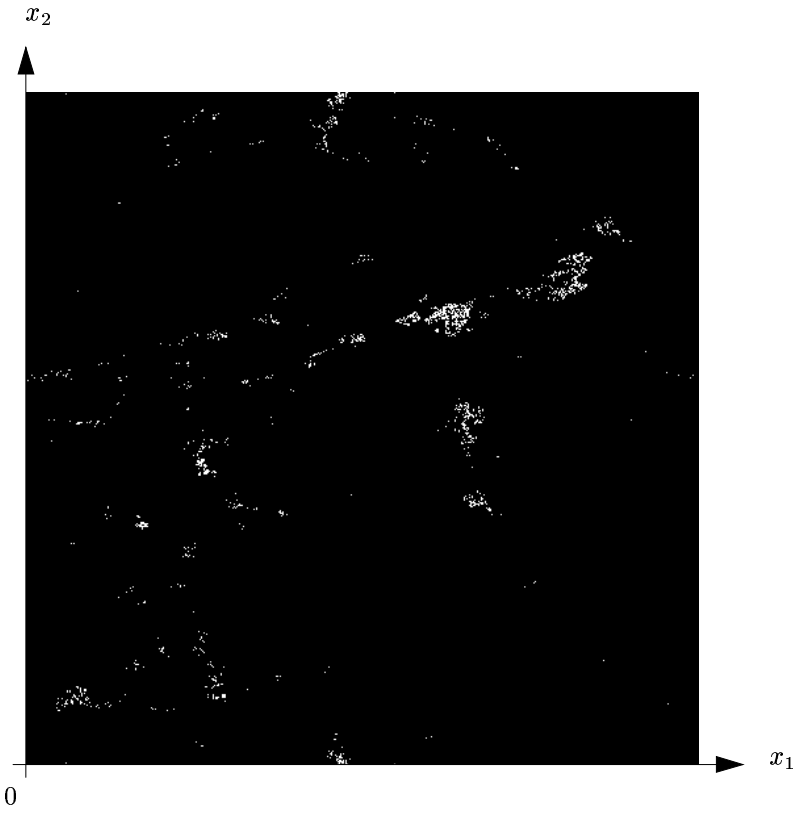

Figure 3. Detected spikes

3.1. Discrete wavelet transform

Not surprisingly, our algorithm for doing this uses once again the wavelet transform, but a discrete one, more suitable for the reconstruction of signals.

Let ψ be a wavelet with compact support as constructed in Ref 10, and let ϕ be the corresponding scaling function. For two-dimensional wavelet decomposition, the usual separable tensor-product construction is used: let us define $\psi_0(x_1, x_2) := \psi(x_1)\psi(x_2)$, $\psi_1(x_1, x_2) := \psi(x_1)\phi(x_2)$ and $\psi_2(x_1, x_2) := \phi(x_1)\psi(x_2)$. Then the functions $\psi_{\epsilon,j,k} : x \mapsto 2^{-j/2}\psi_\epsilon(2^{-j}x - k), j \in \mathbb{Z}, k \in \mathbb{Z}^2, \epsilon \in \{0, 1, 2\}$ form an orthonormal basis of $L^2(\mathbb{R}^2)$.

Using the classical pyramidal algorithm, we compute the full wavelet expansion of our original matrix f. The coefficients obtained are $C_{\epsilon,j,k} := \langle f, \psi_{\epsilon,j,k} \rangle$, $j \in \{1, \ldots, J-1\}$, $k = (k_1, k_2) \in \{0, \ldots, 2^j - 1\}^2$, $\epsilon \in \{0, 1, 2\}$.

3.2. Wavelet coefficients interpolation

We consider that a wavelet coefficient $C_{\epsilon,j,k}$ is *damaged* if at least 25% of the pixels of its 'base square' are damaged, that is, if

$$\sum_{x_1=k_1 2^j}^{(k_1+1)2^j} \sum_{x_2=k_2 2^j}^{(k_2+1)2^j} D(x) \geq 2^{2j-2},$$

is which case we pose $\Delta_{\epsilon,j,k} := 0$, or else $\Delta_{\epsilon,j,k} := 1$.

This ensures that if one pixel is damaged, at least the corresponding smallest-scale wavelet coefficient will be treated.

At this point, we can start 'repairing' the damaged wavelet coefficients, using the following iterative procedure. Let us note $N = \{(-1,-1), (-1,0), (-1,1), (0,1), (1,1), (1,0), (1,-1), (0,-1)\}$.

> for each ϵ, j {
> repeat {
> for each k
> if $\sum_{\kappa \in N} \Delta_{\epsilon,j,k+\kappa} \geq 4$, let $C_{\epsilon,j,k} := \dfrac{\sum_{\kappa \in N} \Delta_{\epsilon,j,k+\kappa} C_{\epsilon,j,k+\kappa}}{\sum_{\kappa \in N} \Delta_{\epsilon,j,k+\kappa}}$
> for each k
> update $\Delta_{\epsilon,j,k}$ (set to 1 the ones we just repaired)
> } until all are repaired (all $\Delta_{\epsilon,j,k} = 1$)
> }

This simply means that we replace a damaged wavelet coefficient by an average of its nearest undamaged neighbors, provided there are at least 4 of them. If this is not the case, this coefficient will wait to be repaired later; it is not difficult to see that, under the hypothesis that there exists a closed contour enclosing the damaged region, this algorithm actually stops (the damaged region will shrink at each step, in a sort of cicatrization process).

3.3. Reconstruction

Once wavelet coefficients are repaired for each j and ϵ, we can do the reconstruction by inverse wavelet transform (Figure 4, with de-periodization).

REMARK 2. *As usual, the choice of the wavelet for the discrete wavelet transform is a trade-off between its regularity, its number of vanishing moments, and the size of its support. A very regular wavelet would be desirable for synthetizing smooth functions, but is not really needed here. On the other hand, if a wavelet has a large support, then the wavelet coefficients may be perturbated by spikes that are far from their 'base square', in which case our algorithm would not list them as damaged. Hence the best results were obtained with a wavelet with a rather short support, as Daubechies' wavelet # 3 on Figure 4.*

4. CONCLUSION

In this paper, we have proposed a method of denoising specifically adapted to the problem of spikes added to non-smooth images. This method consists in first detecting the spikes using the continuous wavelet transform, then removing them by locally interpolating the wavelet coefficients.

ACKNOWLEDGMENTS

This research was supported by AFOSR contract F49620-96-C-0041.

REFERENCES

1. T. Kobayashi and D. A. Shockey, "Applicability of advanced fractography techniques to high-cycle fatigue failure surfaces," Interim technical Report 2, SRI International, Menlo Park, CA, USA, May 1998.
2. T. Kobayashi, D. A. Shockey, C. G. Schmidt, and R. W. Klopp, "Assessment of fatigue load spectrum from fracture surface topography," *Int. J. Fatigue* **19**, pp. S237–S244, 1997.
3. D. A. Shockey, T. Kobayashi, N. Saito, J.-M. Aubry, and A. Grünbaum, "Fractographic analysis of high-cycle fatigue in aircraft engines," tech. rep., SRI International, Jan. 2000.
4. S. Jaffard and Y. Meyer, "Wavelet methods for pointwise regularity and local oscillations of functions," *Mem. Amer. Math. Soc.* **123**, Sep. 1996.
5. S. Mallat, *A wavelet tour of signal processing*, Academic Press, 1998.
6. S. Jaffard, "Functions with prescribed Hölder exponent," *Appl. Comput. Harmonic Anal.* **2**, pp. 400–401, Oct. 1995.
7. N. Saito, "The local Fourier dictionary: A natural tool for data analysis," in *Wavelet Applications in Signal and Image Processing VII*, M. A. Unser, A. Aldroubi, and A. F. Laine, eds., vol. 3813 of *Proceedings of SPIE*, pp. 610–624, 1999.
8. M. V. Wickerhauser, *Adapted Wavelet Analysis from Theory to Software*, A K Peters, Wellesley, MA, 1994.

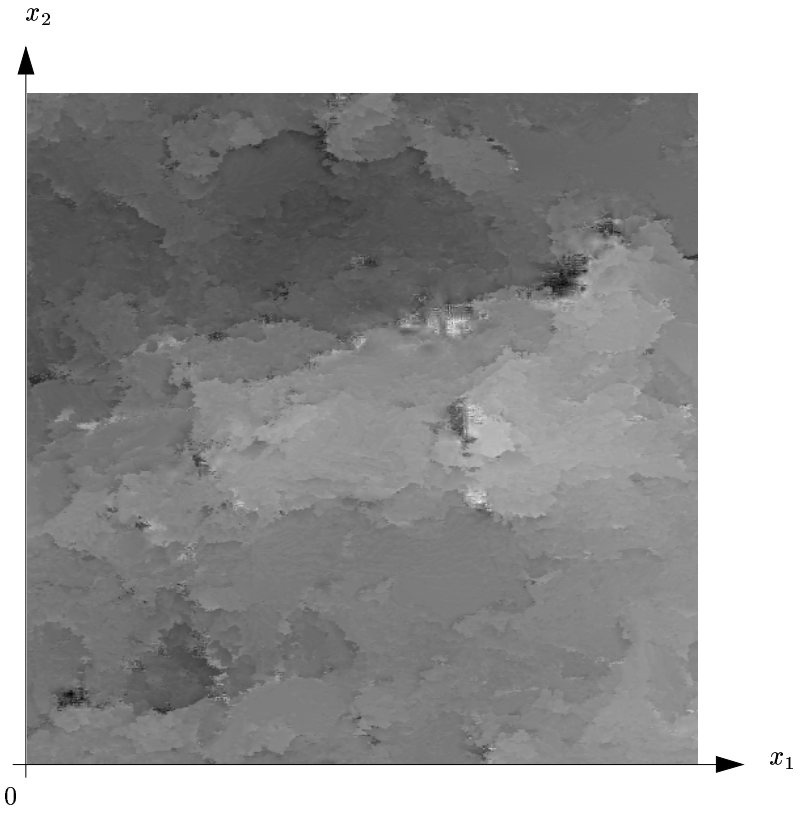

Figure 4. Fractograph after despiking

9. A. Arneodo, E. Bacry, S. Jaffard, and J.-F. Muzy, "Singularity spectrum of multifractal functions involving oscillating singularities," *J. Fourier Anal. Appl.* **4**(2), pp. 159–174, 1998.
10. I. Daubechies, *Ten Lectures on Wavelets*, vol. 61 of *CBMS*, SIAM, 1992.