**Nonlinear Processes
in Geophysics**

An algorithm for detecting layer boundaries in sediments

K. Bube, T. Klenke, and U. Feudel

Institut für Chemie und Biologie des Meeres, Carl von Ossietzky Universität Oldenburg, Postfach 2503, 26111 Oldenburg,
Germany

Abstract. In this paper we present an algorithm based on
wavelet multiscale decomposition, designed to detect lines
of maximal gradients in horizontal direction within two-
dimensional data sets. The algorithm is capable of identi-
fying layer boundaries within sediment profiles, as demon-
strated for artificial as well as two field data sets. Layers are
detected with a good resolution within (i) digital images of
a deep sea sediment core (IODP-expedition 301, core 15H)
and (ii) chemical concentration patterns of recent tidal sedi-
ments (North Sea).

1 Introduction

The interpretation of sedimentary formations is based on the
careful analysis of textural and structural properties. Re-
cently, much progress has been achieved in identifying units
and layers of sedimentary sequences and their internal char-
acteristics as well. This is considerably due to the applica-
tion of methods based on wavelets (Daubechies, 1992; Louis
et al., 1998). Wavelets have been used successfully in the
analysis of seismic (Gunninga and Glinsky, 2006; Capilla,
2006) or gravity data (Fedi et al., 2005; Cooper, 2006), and in
the characterisation of porous media (Sahimi, 2000). How-
ever, there are not many works which use wavelets as a tool
to directly detect sediment layer boundaries. Niebuhr and
Prokoph (1997) used wavelets to detect periodicities in sed-
iment successions from one dimensional data. Maroni et al.
(2001) used a multiscale approach for the detection of deep-
sea subbottom horizons from two dimensional images. This
method looks for the maxima lines in smeared acoustic pro-
filer images.

For many questions it is sufficient to get a rough estima-
tion on the sequence of the layers. However, whenever a

quantitative analysis is needed, it is indispensable to separate
the different layers, to be able to assess statistical measures
for the individual layers. This task is simple as long as the
layer boundaries are just horizontal straight lines or bound-
ary regions can be dropped. But this is often not the case, the
layers can be rippled or even rather complicated in their ge-
ometrical structure. Moreover, the data sets are often small
so that every point is needed to improve the statistics. Thus
an algorithm is required which is able to detect such bound-
aries. Conventional methods from image processing libraries
mostly fail in such cases, because they do not consider addi-
tional information such as the orientation or the curvature of
the boundary line. The standard method in this area is the
one introduced by Canny (1986) and is now part of most im-
age processing toolboxes. Canny's work uses the derivative
of a Gaussian as an optimal filter function which emphasises
large gradients. This method works well for detecting any
gradients in images, but yields lots of false positives as not
every gradient in the image is a layer boundary. One has to
make the definition of a layer boundary more strict.

In this work we present a method which helps to detect
layer boundaries within two-dimensional data sets obtained
from sediment profiles. The data may be high-resolution
photographs or two-dimensional maps of certain features,
e.g. chemical element densities. A change from one layer
to the next one is characterised by a considerable change
of the particular feature over a long contiguous range. This
change can be identified by a strong gradient of the feature.
Our method makes use of the fact that in most applications
the layers are to some extent parallel to a baseline. Natu-
rally this is also the baseline of the data set. This is the ma-
jor difference to conventional edge detection methods (e.g.
Canny, 1986) where no direction information is assumed, but
it turned out that this additional information is very helpful
in order to avoid lots of false positives. The presented algo-
rithm is based on a two-dimensional multiscale decomposi-
tion (Mallat, 1989a) of the data. More or less horizontal lines

Correspondence to: K. Bube
(bube@icbm.de)

of maximal gradients are searched within the coarse grained data. Afterwards the lines are transferred back to the finer scales. After applying this procedure it is possible to detect the different layers and subsequently to analyse the properties of every single layer.

First we present the proposed algorithm in detail by means of an artificially created profile. We evaluate this profile where the layer boundaries are known and compare the expected versus the calculated result. We proceed by applying our method to a core taken from the Pacific deep sea floor, where the layer boundaries are quite smooth and regular. Afterwards we analyse a micro-sequence of a profile taken at Mellum Island in the North Sea. This illustrates the capabilities of the algorithm when applied to sediments with a clustered structure where the boundaries are very irregular. At the end we summarise the results and draw our conclusions.

2 Method

The basic idea of our method is that we make use of a modified multiscale approach which was originally developed by Maroni et al. in 2001 to automate the search for layer boundaries in our sediment profiles. The algorithm is based on a wavelet decomposition of the underlying image in various scales (Mallat, 1989a,b). While in the original work by Maroni et al. the aim was to find longest connected lines of similar height within greyscale images, we are looking for maximal gradients. Furthermore, we use smoothing techniques in order to suppress noise influences and a modified line following algorithm which manifests the assumption that the orientation of the layer boundaries does not deviate too much from the baseline orientation. This has proven to help avoiding many false positives for our application.

2.1 Theoretical background

The multiscale decomposition method which is the heart of our proposed algorithm allows a signal $f(x, y)$ to be decomposed into two parts: an approximation part $a_s(x, y)$ and a detail part $d_s(x, y)$ at a certain scale s. The detail part contains all information of the signal at s, while the approximation part contains all information of f at smaller scales. The original signal can be retrieved by adding both

$$f(x, y) = a_s(x, y) + d_s(x, y) . \qquad (1)$$

$a_s(x, y)$ can further be decomposed and more and more details can be removed from the signal and a coarse representation of the image is obtained

$$f(x, y) = a_{s-n}(x, y) + d_s(x, y) + \cdots + d_{s-n}(x, y) . \qquad (2)$$

Actually in two dimensions the detail coefficients $d_s(x, y)$ are split into three parts d_s^h, d_s^v and d_s^d which hold the information contained in horizontal, vertical and diagonal direction. However, this is not relevant for our application and

we summarise this as the detail part. For our analysis we are only interested in the approximation part.

The decomposition algorithm works the following way: Let $f(x, y) \in \mathbf{L}^2(\mathbb{R}^2)$ be the measured image. $\mathbf{L}^2$ is the vector space of measurable, square-integrable functions. There exists a "scaling function" $\phi(x) \in \mathbf{L}^2(\mathbb{R})$, such that if $\phi_{2^j}(x) = 2^j \phi(2^j x)$ is the dilatation of $\phi(x)$ by 2^j, $j \in \mathbb{Z}$, then $(\sqrt{2^{-j}}\phi_{2^j}(x - 2^{-j}n)\phi_{2^j}(y - 2^{-j}m))_{(n,m) \in \mathbb{Z}^2}$ is an orthonormal basis of the vector space $\mathbf{A}_{2^j}$ of all possible approximations at scale 2^j of the functions in $\mathbf{L}^2$ (Mallat, 1989a). We can approximate the signal $f(x, y)$ with ϕ_{2^j} by calculating the inner products

$$
\begin{aligned}
a_{2^j}(n, m) = & \\
& \langle f(x, y), \phi_{2^j}(x - 2^{-j}n)\phi_{2^j}(y - 2^{-j}m)\rangle_{(n,m) \in \mathbb{Z}^2} \\
= & \iint_{-\infty}^{+\infty} f(x, y)\phi_{2^j}(x - 2^{-j}n)\phi_{2^j}(y - 2^{-j}m) \, dx dy .
\end{aligned}
$$
$$(3)$$

A similar theorem exists for the detail part at scale 2^j. It can be shown that for every scaling function $\phi(x)$ there exists a corresponding orthogonal wavelet $\psi(x) \in \mathbf{L}^2(\mathbb{R})$, where $(\sqrt{2^{-j}}\psi_{2^j}(x - 2^{-j}n))_{n \in \mathbb{Z}}$ forms an orthogonal basis in the space of signal details.

Let $\mathbf{A}_{2^j}$ be the vector space of signal approximations at scale 2^j, then

$$\mathbf{A}_{2^{j+n}} \subset \ldots \subset \mathbf{A}_{2^{j+1}} \subset \mathbf{A}_{2^j} . \qquad (4)$$

This means that the image may be decomposed into coarser and coarser parts as the finer scales are successively removed. This is the property we want to exploit for our purposes. For our analysis we use the well known Haar wavelet

$$\psi(t) = \begin{cases} 1 & 0 \leq t < \frac{1}{2} \\ -1 & \frac{1}{2} \leq t < 1 \\ 0 & \text{otherwise} , \end{cases} \qquad (5)$$

whose corresponding scaling function is

$$\phi(t) = \begin{cases} 1 & 0 \leq t \leq 1 \\ 0 & \text{otherwise} \end{cases} \qquad (6)$$

(Louis et al., 1998). This wavelet is also known as the Daubechies D2 wavelet as it is a special case of the Daubechies wavelet family and is orthogonal to polynomials up to second order (Daubechies, 1992). Wavelets with higher vanishing moments may also be used for our method and may yield better results in some applications. However, this was not the case with our data sets, so we decided to keep the simplest wavelet possible.

2.2 The algorithm

The main idea of the algorithm is to look first at a coarse grained instance of the original image, search for some desired property and then propagate it back again to the finer
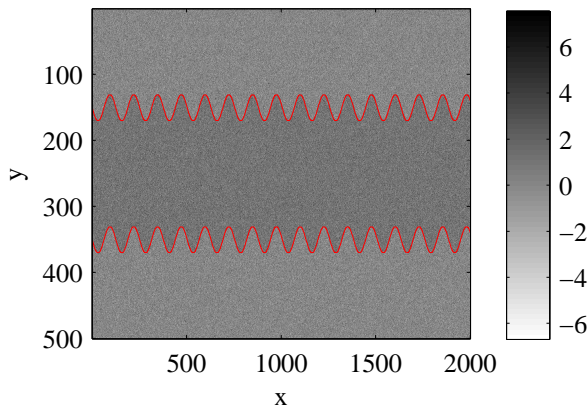
Fig. 1. Artificial example layers. The boundaries are sines. In the two outer layers to each pixel the value 0 is assigned as the characteristic feature, while in the middle layer this value is elevated by 1. Afterwards Gaussian random numbers with a variance of 2 and zero mean are added to the image. The red lines mark the theoretical, undisturbed boundaries.

Fig. 2. Absolute value of the gradient in the y-direction of the decomposed images. From upper left to lower right decomposition levels 4 to 1 are shown. The size of the moving average window is 2 in x- and 1 in y-direction.

scales. The property we are interested in are distinguished gradients more or less parallel to the image baseline. In order to demonstrate the algorithm, we produce an artificial sediment profile. We generate an empty image filled with zeros, add two sinusoidal layer boundaries and fill the space in between them with ones. Afterwards we add random Gaussian noise with a variance of 2 to the whole image. Thus, the characteristic feature of the layers are their mean values. The result is shown in Fig. 1.

2.2.1 Finding the line seeds

1. As can be seen in Fig. 1, the raw image is very irregular. Although it is possible to distinguish brighter and darker layers by eye, it is very difficult for a computer to distinguish different regions. This is a common problem also with sediment data. Fortunately, the disturbances only play a role at small scales, so we use the multiscale decomposition to remove them.

   We start the algorithm by decomposing the measured image $f(x, y)$ according to the Mallat algorithm described in Sect. 2.1. According to Eq. (3) this gives us a series of images $a_{2^d}(x, y)$, $d=0 \dots l$, where l is the maximum decomposition level ($d=0$ is the original image). l has to be chosen by the user. Usually the finiteness of the data sets tight limits on what is a useful coarse graining.

2. We define a layer boundary as a more or less horizontal curve where the gradients in vertical y-direction are maximal. Therefore, we do not look at the decomposed image $a_{2^l}(x, y)$ itself, but at the image of absolute values of gradients in y-direction
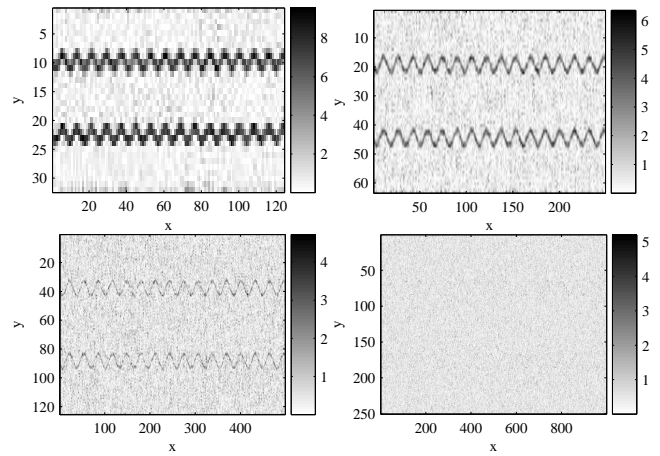
$$\Delta a_{2^l}(x, y) = |a_{2^l}(x, y) - a_{2^l}(x, y + 1)|. \tag{7}$$

In Fig. 2 this is shown for our artificial example. It can be seen that the width and height of the image is divided by 2 for every coarse graining step.

3. Often the approximation image is still quite irregular. Thus in the next step the decomposed gradient image at level l may be smoothed by a moving average filter

$$\Delta h(x, y) = \frac{1}{MN} \sum_{j=y}^{M+y-1} \sum_{i=x}^{N+x-1} \Delta a_{2^l}(i, j), \tag{8}$$

where N and M are the sizes of the moving average window. These parameters have to be chosen by the user. The results have to be harmonised with the expected results by adjusting N and M. With our used data sets this was possible within a few iterations. Unsuitable values often result in sudden high jumps in the layer boundaries. The right configuration yields quite smooth boundaries with only a few jumps.

4. Our aim is to find the k strongest gradient lines which are more or less parallel to the x-axis of the image and which are as long as possible. This number of lines k we search for has to be chosen in advance. As the lines do not have to be exactly parallel to the x-axis, we calculate the deviation of each column at position x from the parallel as the cross correlation to the first column

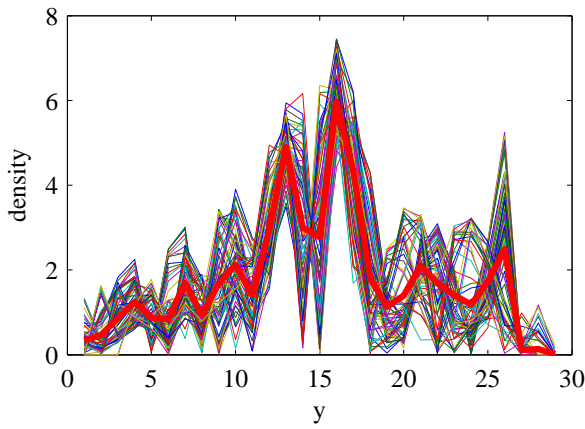$$\xi(\tau(x)) = \sum_y \Delta h(1, y) \Delta h(x, y + \tau). \tag{9}$$

Fig. 3. Cross sections along the y-axis of Δh. The lines are shifted by the lag τ which yields the maximal correlation coefficient ϕ between the first and the concerned line. The thick red line is the mean of all lines.
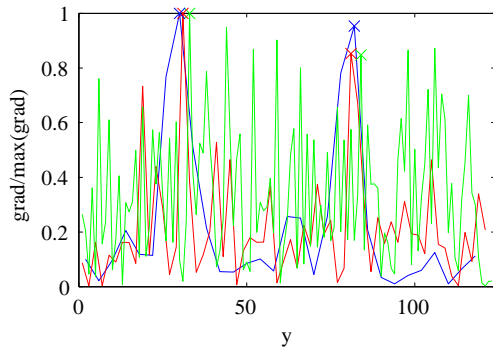

Fig. 4. Maximum gradients of a column within various coarseness levels. The blue line denotes column 30 of the gradient image at level 4, the red line is column 60 at level 3 and the green line column 120 at level 2. The crosses mark the two tracked maxima. The position of the points on the abscissa is multiplied with 2^j, where j is the scale. This way points from different scales neighbour each other. The curves are normalised so that the maximum is 1, in order to make them comparable.

To get the main layer boundaries we take the average over all gradients shifted by $\tau(x)$ where ξ is maximal

$$\Delta\tilde{h}(y) = \frac{1}{N}\sum_{x}^{N}\Delta h(x, y + \tau(x)).\qquad(10)$$

Let the k highest maxima in $\Delta\tilde{h}(y)$ be at y_i, with $i=1\ldots k$. Then our layer boundaries in the coarsest image are at $(x, y_i+\tau(x))$. To demonstrate this procedure, a cross section along the y-axis is shown in Fig. 3. The shifted profiles and the resulting mean value are displayed.
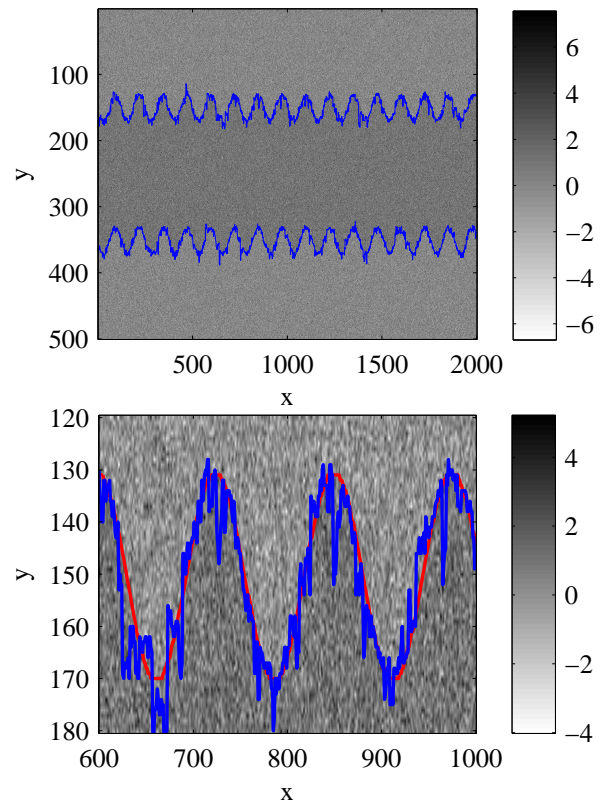

Fig. 5. Upper panel: The blue lines are the layer boundaries reconstructed by the algorithm from the noisy artificial sediments. Lower panel: a magnification of the upper boundary. The red line indicates the theoretical boundary and the blue line the one the algorithm calculated.

2.2.2 Propagating the seed lines to the finer images

The propagation to the finer scales is quite straightforward. In the previous subsection we calculated the initial lines to be at $(x, y_i+\tau(x))$ at resolution level l. In level $l-1$ the width and height of the image doubles for the Haar wavelet and the lines from the coarser level get smeared from (x, y) to $(2x, 2y)$. The upscaling happens now as follows.

1. We calculate the gradients in y-direction (see Eq. 7) of $a_{2^{l-n}}(x, y)$, where $n=1\ldots l$.

2. Then the result is smoothed with the moving average procedure according to Eq. (8).

3. Let $(x, y_i+\tau(x))$ with $i=1\ldots k$ be the layer boundaries at resolution $l-n+1$. As mentioned above the upscaled boundary at $l-n$ gets smeared around $(2x, 2(y_i+\tau(x)))$. Therefore we look for the maxima in a band at $(2x, 2(y_i+\tau(x))+\epsilon)$ and $(2x+1, 2(y_i+\tau(x))+\epsilon)$ with $\epsilon=-1, 0, 1$. Looking at $2x+1$ is necessary as the size of the upscaled image is doubled. These are the positions of the boundaries at resolution $l-n$.
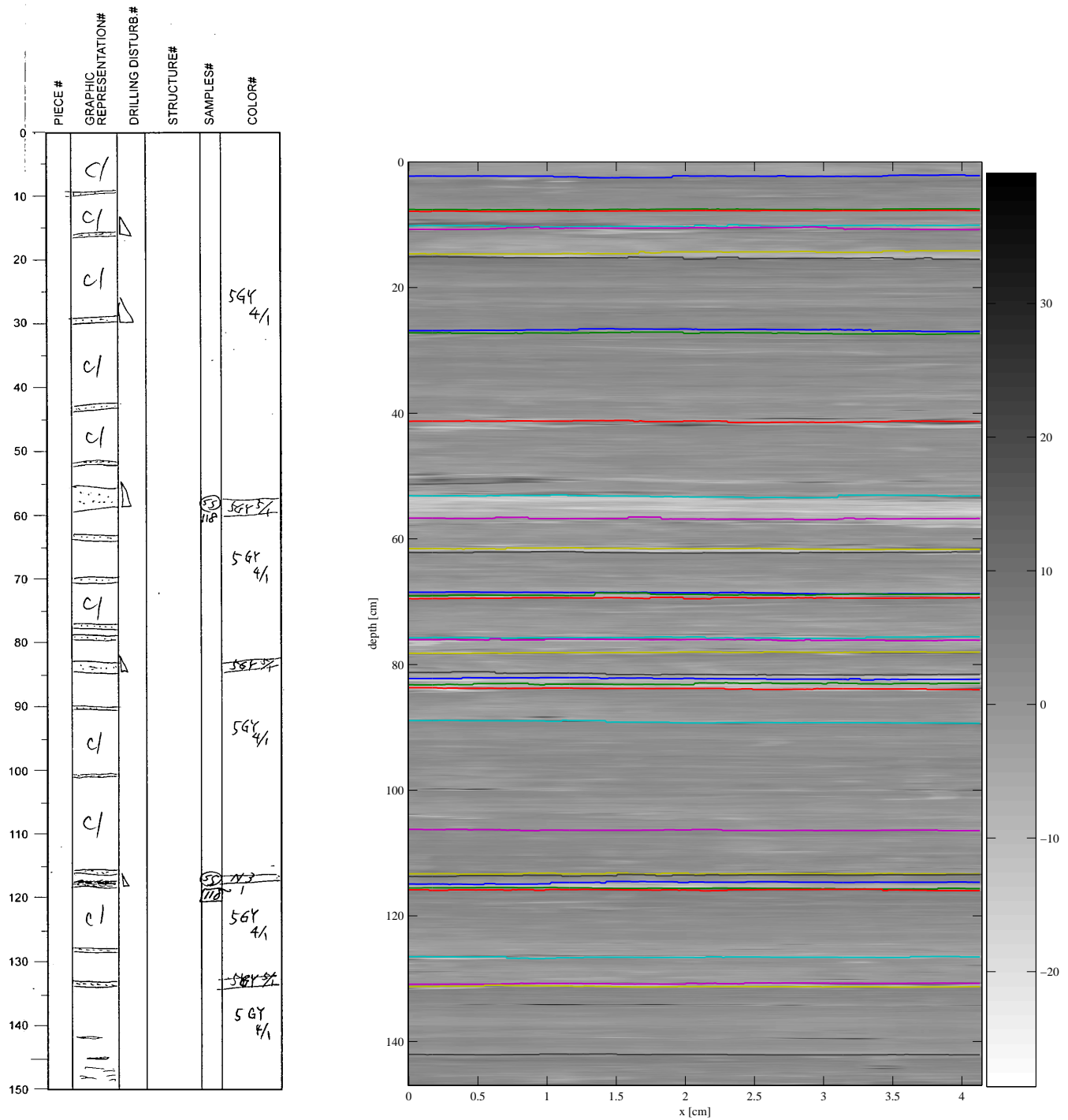
Fig. 6. Left: the visual core description from the expedition log. (Picture courtesy of IODP.) Right: The layer boundary lines found by the algorithm. A moving average smoothing of 20 in x-direction and 3 in y-direction was used. The number of lines is 35.

4. The procedure has to be repeated until resolution level 0 is reached. Figure 4 shows the tracking of two maxima throughout the upscaling process.

The layers reconstructed by the detection algorithm are displayed in Fig. 5. It can be seen that despite the bad sig-

nal to noise ratio of the example, found boundaries follow very good the theoretical layer boundary. As shown in Fig. 2 the multiscale decomposition successfully removes the noise which mainly acts on the small scales, but the large structures, which are the ones we are interested in remain. When
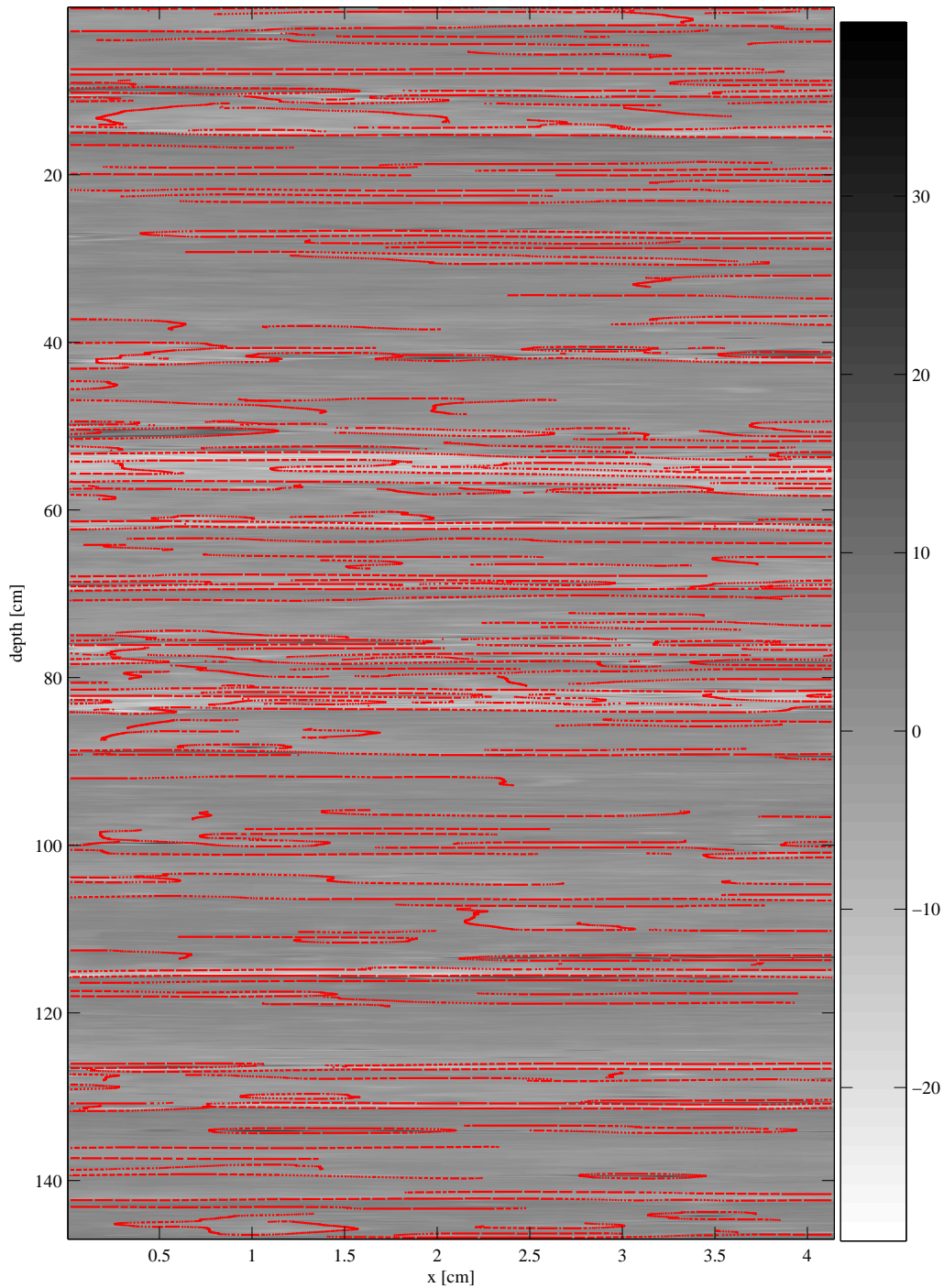
Fig. 7. Layer boundaries found by the Canny edge detection algorithm. The standard deviation for the Gaussian filter was set to 50.

going up to the finer scales the fluctuations within the image have also an impact on the boundary which becomes visible in the magnification.

3   Borehole data

As an application for the algorithm introduced in the previous section, we analyse samples from deep-sea borehole cores. The data are taken from the Integrated Ocean Drilling

Program (IODP) expedition 301[1]. This expedition gathered core samples from the eastern flank of the Juan de Fuca Ridge west of North America (Expedition 301 Scientists, 2005). The section analysed by us is from core 15H, Sect. 3.

We convert the high-resolution colour JPEG image from the database to a greyscale image. The resulting image matrix has $14\,700 \times 433$ points. Systematic trends are removed by subtracting the mean from each column.

The image is decomposed down to the fourth level and smoothed with a moving average window of size 20×3. This size turned out to yield the smoothest lines and eliminates high jumps which occur at smaller window sizes. We search for 35 boundaries. The resulting boundaries found by the algorithm are shown in Fig. 6. By comparing the algorithmic result on the right with the lines drawn by the ship board party on the left it can be seen that they match very well. Nearly every boundary found by visual inspection is also automatically detected by the algorithm. Even the finer structures are well resolved. In between 0 and 10 cm of depth two false positives are found, which are likely due to artifacts in the image or from the cutting of the core. The boundary at 100 cm is not found, probably as it is too badly resolved on the photograph.

The result of the proposed algorithm looks even better if it is compared with the result of the Canny edge detector displayed in Fig. 7. The standard detector is clearly unsuitable for the application as it finds much too many boundaries. As it makes no assumption on the orientation of the layers, it detects every contour within the image. This is what the algorithm was designed for, but this is clearly undesirable in this particular application. Also adjusting the standard deviation of the Gaussian filter does not improve the result significantly.

4 Experimental data from tidal sediments

Now we demonstrate the capabilities of the algorithm to find boundaries of complicated geometrical structure. To this end we analyse samples from a barrier island (Mellum Island) in the Wadden Sea in the German part of the southern North Sea. This island is located at $53°43'16''$ N, $8°8'58''$ E. The samples were taken from lower supratidal areas and consist of porous siliciclastic sediments with high biological activity. The laminated sediments are nerved with microbial mats (Gerdes et al., 1985; Block et al., 1991).

Small pieces were sub-sampled and surveyed as thin polished films under a scanning electron microscope equipped with an energy dispersive x-ray analysis device. This gives us a spatial image of the densities of various chemical elements, i.e. aluminium, calcium and silicon. The element densities give insight into the composition of the sediment. The succession of various layers indicates different processes that formed it.
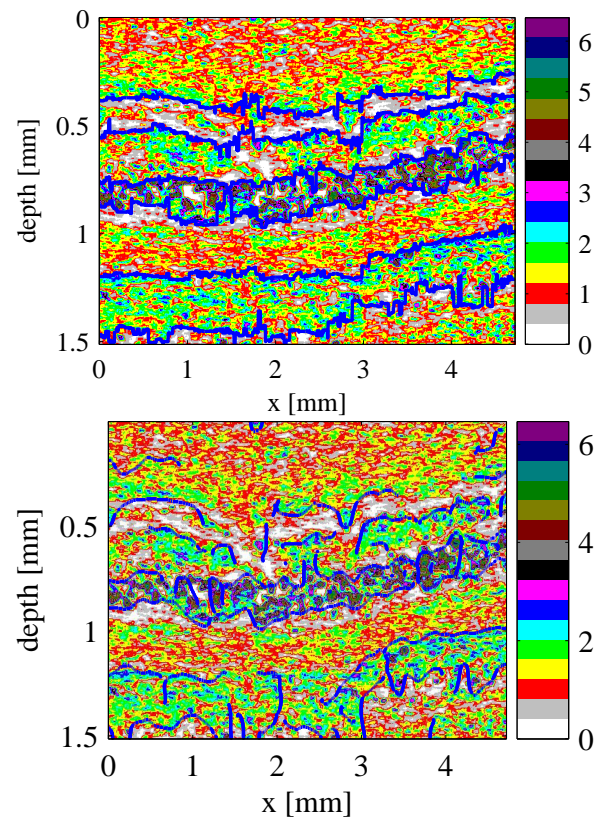
---

[1] http://iodp.tamu.edu/scienceops/expeditions/exp301.html

Fig. 8. Upper panel: Densities for calcium smoothed by a moving average window of size 20 in x-direction and 3 in y-direction. In blue are the calculated layer boundaries. Lower panel: The same but the layers are searched by the Canny edge detector. The standard deviation of the filter function was set to 15.

Out of the three measured chemical elements calcium shows the most distinguished layer structure. We set the number of boundaries to be found to 6 and got the lines displayed in Fig. 8, upper panel. The boundaries found in the image match very well to what one would expect from visual inspection.

The result for the Canny edge detector is shown in Fig. 8, lower panel. For this data set the method performs much better than for the borehole dataset, but again the result of the multiscale algorithm is better. The boundary of the calcium rich zone is detected quite well, but the boundary line is broken at many places. The boundaries of the other zones are indicated but even more discontinuous.

It is now possible to compare the properties of the calcium layers to the layers of the other elements. Therefore, we transfer the boundaries obtained from the analysis of calcium to the measurements of the other elements. This is shown in Fig. 9. For calcium we can see that the fourth layer (counted from top to bottom) consists of large clusters of calcium. For silicon and aluminium this is a depletion zone. In the sixth layer accumulations of silicon and aluminium can be found.
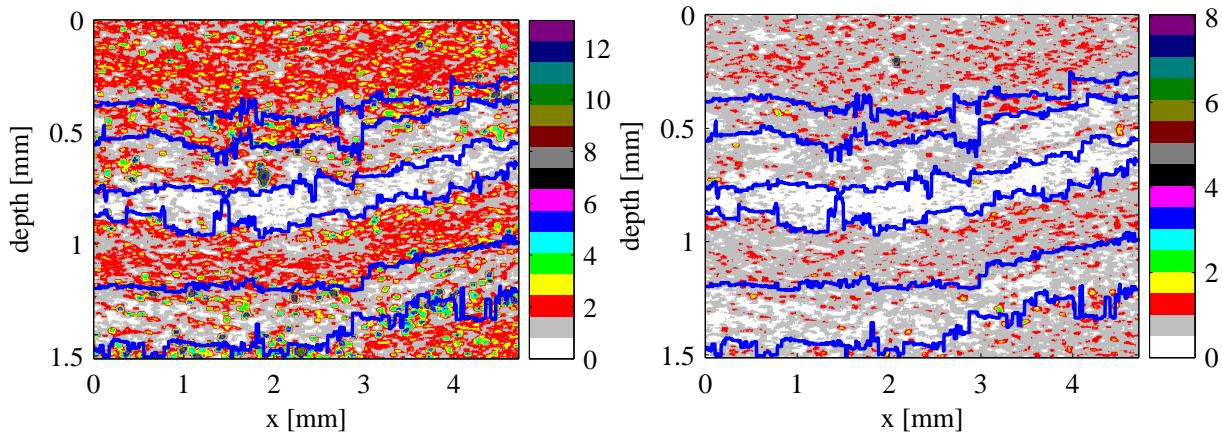
Fig. 9. Left: image of silicon densities. Right: image of aluminium densities. The size of the moving average window was the same as for calcium (see Fig. 8).

Table 1. Mean densities $\langle \rho \rangle$ and standard deviation σ within the different layers.

Layer	Si		Ca		Al	
	$\langle \rho \rangle$	σ	$\langle \rho \rangle$	σ	$\langle \rho \rangle$	σ
1	1.902	0.644	1.245	0.417	0.815	0.272
2	2.057	1.119	0.880	0.497	0.773	0.237
3	1.536	1.102	1.420	0.679	0.616	0.216
4	0.893	0.389	2.462	1.228	0.457	0.144
5	1.707	0.634	1.180	0.526	0.759	0.225
6	1.688	1.130	1.636	0.603	0.648	0.232
7	2.444	1.437	1.191	0.578	0.760	0.253

In general one can see that the lines clearly divide regions of different characteristics in terms of element densities.

Table 1 lists the mean measured densities $\langle \rho \rangle$ and the standard deviation σ for the found layers. It can be seen that the standard deviations are quite high compared to the mean densities for all layers. This is because of small clusters of very high densities which are existent in all layers. Here a cluster size analysis and the comparison of density distributions are more suitable to get more pronounced differences between the layers. In Fig. 10 the density distributions for the different layers are shown. We do not go into interpretation details here, but it can be seen that the characteristics within the layers are clearly different. The obtained distributions make it possible to develop and fit theoretical process models.

5 Conclusions

In this article we present an algorithm, which is able to discriminate layers of different characteristics within sediments. Our application to artificial and real experimental systems yields very good agreements between expected and calcu-
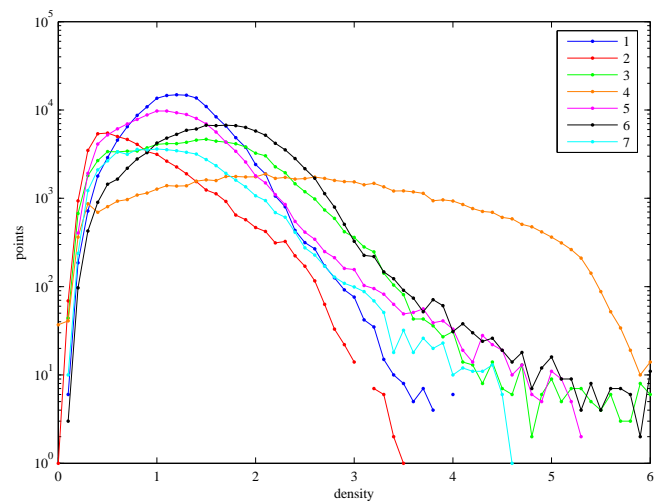

Fig. 10. Distribution of calcium densities within the calculated layers.

lated results. The algorithm has four degrees of freedom which have to be chosen by the user: the maximum decomposition level, the number of layer boundaries and the width and height of the moving average window. The maximum decomposition level is strongly dependent on the amount of the data measured. For our application a number of 4 gives satisfying results and a higher number would have led to unacceptable small images. If the number of layers is not too high it sometimes can be guessed by visual inspection, otherwise one has to play a bit by adding more and more boundaries. The size of the moving average window is a bit more tricky and has to be found by trying different sizes, but in practice this has proven to be not too difficult.

We have shown that the algorithm significantly outperforms the results of a standard edge detection method.

The algorithm may be a helpful tool for detecting layer boundaries as a prerequisite for a quantitative analysis of sediment profiles. Extensions of this algorithm are possible by using other wavelets as the basis of the decomposition.

References

Block, A., von Bloh, W., Klenke, T., and Schellnhuber, H. J.: Multifractal Analysis of the Microdistribution of Elements in Sedimentary Structures Using Images From Scanning Electron Microscopy and Energy Dispersive X Ray Spectroscopy, J. Geophys. Res., 96, 223–230, 1991.

Canny, J.: A Computational Approach to Edge Detection, IEEE Trans. Patt. Anal. Mach. Intell., 8, 679–698, 1986.

Capilla, C.: Application of the Haar wavelet transform to detect microseismic signal arrivals, J. Appl. Geophys., 59, 36–46, 2006.

Cooper, G.: Interpreting potential field data using continuous wavelet transforms of their horizontal derivatives, Comput. Geosci., 32, 984–992, 2006.

Daubechies, I.: Ten Lectures on Wavelets, vol. 61 of CBMS–NSF Regional Conference Series in Applied Mathematics, SIAM Publications, Philadelphia, 1992.

Expedition 301 Scientists: Expedition 301 summary, in: Proc. IODP, edited by: Fisher, A., Urabe, T., Klaus, A., and the Expedition 301 Scientists, vol. 301, Integrated Ocean Drilling Program Management International, Inc., College Station TX, doi:10.2204/iodp.proc.301.101.2005, 2005.

Fedi, M., Paoletti, V., and Rapolla, A.: The role of multilevel data in potential field interpretation, Comput. Geosci., 31, 681–688, 2005.

Gerdes, G., Krumbein, W. E., and Reineck, H.-E.: The depositional record of sandy, versicolored tidal flats (Mellum island, southern North Sea), J. Sediment. Petrol., 55, 265–278, 1985.

Gunninga, J. and Glinsky, M.: Wavelet extractor: A Bayesian well-tie and wavelet extraction, Comput. Geosci., 32, 681–695, 2006.

Louis, A. K., Maaß, P., and Rieder, A.: Wavelets, Teubner, Stuttgart, 1998.

Mallat, S. G.: A Theory for Multiresolution Signal Decomposition: The wavelet representation, IEEE Trans. Pattern Anal. Mach. Intell., 11, 674–693, 1989a.

Mallat, S. G.: Multiresolution Approximations and Wavelet Orthonormal Bases of $L^2(R)$, Trans. Amer. Math. Soc., 315, 69–87, 1989b.

Maroni, C.-S., Quinquis, A., and Vinson, S.: Horizon Picking on Subbottom Profiles Using Multiresolution Analysis, Digital Signal Processing, 11, 269–287, 2001.

Niebuhr, B. and Prokoph, A.: Periodic-cyclic and chaotic successions of Upper Cretaceous (Cenomanian to Campanian) pelagic sediments in the North German Basin, Cretaceous Research, 18, 731–750, 1997.

Sahimi, M.: Fractal-wavelet neural-network approach to characterization and upscaling of fractured reservoirs, Comput. Geosci., 26, 877–905, 2000.