1

Beyond wavelets: New image representation paradigms

Hartmut Führ, Laurent Demaret, Felix Friedrich

GSF National Research Center for Environment and Health
IBB Institute of Biomathematics and Biometry
D-85764 Neuherberg

Abstract

It is by now a well-established fact that the usual two-dimensional tensor product wavelet bases are not optimal for representing images consisting of different regions of smoothly varying grey values, separated by smooth boundaries. The chapter starts with a discussion of this phenomenon from a nonlinear approximation point of view, and then proceeds to describe approaches that have been suggested as a remedy. The methods can be sorted roughly into two groups: Adaptive geometry-based approaches such as wedgelets and related constructions on one hand, and directional frames, such as curvelets or ridgelets, on the other. We discuss wedgelets and curvelets in more details, as representatives of the different branches. These systems are first described in the continuous setting, and their construction is motivated by a discussion of their nonlinear approximation properties. We then present digital implementations of the schemes. For wedgelets and related transforms, we present a new method which results in a significant speedup, in comparison to preexisting implementations. We also give a short description of the contourlet approach to discrete curvelets. In the last section, we present the results of nonlinear approximation experiments, comparing wedgelets, contourlets and wavelets, and comment on the potential of the new techniques for image coding.

Introduction

Despite the huge success of wavelets in the domain of image compression, the failure of two-dimensional multiresolution wavelets when dealing with images of the cartoon class, i.e., images consisting of domains of smoothly varying grey values, separated by smooth boundaries, has been noted repeatedly. This failure can be discerned visually, in the form of characteristic compression artefacts, but it can also be derived theoretically. In this chapter we review some of the constructions that were proposed as a remedy to this problem. We focus on two constructions, *wedgelets* [16] and *curvelets* [6]. Both systems stand for larger classes of image representation schemes; let us just mention *ridgelets* [4], *contourlets* [12, 14], *beamlets* [17], *platelets* [35] and *surflets* [7] as close relatives.

The chapter starts with a discussion of the shortcomings of wavelet orthonomal bases. The reason for expecting good approximation rates for cartoon-like images is the observation that here the information is basically contained in the edges. Thus, ideally, one expects that smoothness of the boundaries should have a beneficial effect on approximation rates. However, the tensor product wavelets usually employed in image compression do not adapt to smooth boundaries, due to the isotropic scaling underlying the multiresolution scheme. The wedgelet scheme tries to overcome this by combining adaptive geometric partitioning of the image domain with local regression on the image segments. A wedgelet approximation is obtained by minimizing a functional that weighs model complexity (in the simplest possible case: the number of segments) against approximation error. By contrast, the curvelet approach can be understood as a directional filterbank, designed and sampled so as to ensure that the system adapts well to smooth edges (the key feature here turns out to be *hyperbolic scaling*) while at the same time providing a *frame*, i.e., a system of building blocks with properties similar to an orthonormal bases. Here nonlinear approximation is achieved by a simple truncation of the frame coefficients.

After a presentation of these constructions for the continuous setting, we then proceed with a description of methods for their digital implementation. We sketch a recently developed, particularly efficient implementation of the wedgelet scheme, as well as the contourlet approach to curvelet implementation, as proposed by Do and Vetterli [12, 13, 14].

In the last section, we present some numerical experiments to compare the nonlinear approximation behavior of the different schemes, and contrast the theoretical approximation results to the experiments. We close by commenting on the potential of wedgelets and curvelets for image coding. Clearly, the nonlinear approximation behavior of a scheme can only be used as a first indicator of its potential for image coding. The good ap-

proximation behavior of the new methods for *small* numbers of coefficients reflects their ability to pick out the salient geometric features of an image rather well, which could be a very useful property for hybrid approaches.

1.1 The Problem and Some Proposed Solutions

Besides the existence of fast decomposition and reconstruction algorithms, the key feature that paved the way for wavelets is given by their ability to effectively represent discontinuities, at least for one dimensional signals. However, it has been observed that the tensor-product construction is not flexible enough to reproduce this behaviour in two dimensions. Before we give a more detailed analysis of this failure, let us give a heuristic argument based on the wavelet coefficients displayed in Figure 1.1. Illustrations like this are traditionally used to demonstrate how wavelets pick salient (edge) information out of images. However, it has been observed previously (e.g. [13]), that Figure 1.1 in fact reveals a weakness of wavelets rather than a strength, showing that wavelets detect isolated *edge points* rather than *edges*. The fact that the edge is smooth is not reflected adequately; at each scale j the number of significant coefficients is proportional to 2^j times the length of the boundary, regardless of its smoothness.

A more quantitative description of this phenomenon can be given in terms of the *nonlinear approximation error*. All theoretical considerations below refer to a Hilbert space of (one- or two-dimensional) signals, and the approximation performance is measured by the Hilbert space norm. In the continuous domain setting, the Hilbert space will be $\mathrm{L}^2(\mathbb{R}^d)$, the space of functions f on $\mathbb{R}^d$ that are square-integrable with respect to the usual Lebesgue measure, with scalar product

$$\langle f, g\rangle = \int_{\mathbb{R}^d} f(x)\overline{g(x)}dx \quad ,$$

and norm $\|f\|_2 = \langle f, f\rangle^{1/2}$. In the discrete domain setting, the signals under consideration are members of $\ell^2(\mathbb{Z}^d)$, the square-summable sequences indexed by $\mathbb{Z}^d$. To keep the notation in the following arguments simple, we let C denote a constant that may change from line to line.

Now suppose we are given an orthonormal basis $(\psi_\lambda)_{\lambda\in\Lambda}$ of the signal Hilbert space. For a signal f and $N \geq 0$ we let $\epsilon_N(f)$ denote the smallest possible squared error that can be achieved by approximating f by a linear approximation of (at most) N basis elements, i.e.

$$\epsilon_N(f) = \inf \quad \{ \quad \|f - \sum_{\lambda\in\Lambda'} \alpha_\lambda\psi_\lambda\|^2 \\ : \ \Lambda' \subset \Lambda \text{ with } |\Lambda'| = N, (\alpha_\lambda) \in \mathbb{C}^\Lambda\} \quad .$$

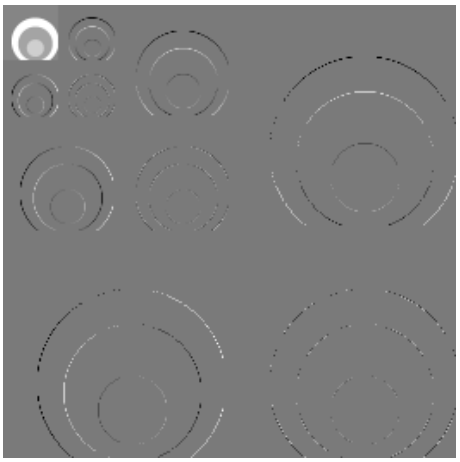

Figure 1.1: Wavelet coefficients of an image with smooth edges. The detail images are renormalised for better visibility.

The study of the nonlinear approximation error can be seen as a precursor to rate-distortion analysis; for more details on nonlinear approximation and its connections to compression we refer to the survey given in [29], or to Chapter IX in Mallat's book [23]. Since we started with an orthonormal basis, the approximation error is easily computed from the expansion coefficients $(\langle f, \psi_\lambda \rangle)_{\lambda \in \Lambda}$, by the following procedure that relates the asymptotic behaviour of the approximation error to the decay behaviour of the coefficients: Reindex the coefficients to obtain a sequence $(\theta_m)_{m \in \mathbb{N}}$ of numbers with decreasing modulus. Then the Parseval relation associated to the orthonormal basis yields

$$\epsilon_N(f) = \sum_{m=N+1}^{\infty} |\theta_m|^2 \quad . \tag{1.1}$$

Let us now compare the approximation behaviour of one- and two-dimensional wavelet systems. We only give a short sketch of the argument, which has the purpose to give a closer description of the dilemma surrounding two-dimensional wavelets, and to motivate the constructions designed as remedies. Generally speaking, the mathematics in this chapter will be held on an informal level.

Let $f : [0,1] \to \mathbb{R}$ be a bounded function that is piecewise n-times continuously differentiable, say outside a finite set $S \subset [0,1]$ of singularities; we will call these functions **piecewise $\mathbf{C^n}$**. Let $(\psi_{j,k})_{j \in \mathbb{Z}, k \in \mathbb{Z}}$ be a wavelet orthonormal basis consisting of compactly supported functions with n van-

ishing moments. Recall, e.g. from [23], that the wavelet orthonormal basis is obtained from a suitable mother wavelet ψ via

$$\psi_{j,k}(x) = 2^{j/2}\psi(2^j x - k) \quad .$$

If ψ is supported in the interval $[-r, r]$, then $\psi_{j,k}$ is supported in $[2^{-j}(k - r), 2^{-j}(k + r)]$.

The structure of the following argument will be encountered several times in this chapter. Basically, we differentiate between two cases, depending whether the support of the wavelet $\psi_{j,k}$ contains a singularity or not. If it does contain a singularity, we employ fairly rough estimates. In the other case, the smoothness of the signal and the oscillatory behaviour of the wavelet combine to yield the appropriate decay behaviour. The overall estimate of the decay behaviour then crucially relies on the fact that the first case does not occur too often.

In the one-dimensional setting, on each dyadic level $j \geq 0$, corresponding to scale 2^{-j} the number of positions k such that the support of $\psi_{j,k}$ contains a singularity of f is fixed (independent of j), and for these coefficients we can estimate

$$\begin{aligned} |\langle f, \psi_{j,k}\rangle| &= \left| \int_{\mathbb{R}} f(x)\overline{\psi_{j,k}(x)}dx \right| \\ &\leq \sup_{x\in\mathbb{R}}|f(x)| \int_{\mathbb{R}} 2^{j/2}|\psi(2^j x - k)|dx \\ &= \|f\|_\infty 2^{-j/2} \int_{\mathbb{R}} |\psi(x)|dx \\ &= \|f\|_\infty \|\psi\|_1 2^{-j/2} \quad . \end{aligned}$$

Likewise, for $j < 0$, the number of k such that the support of $\psi_{j,k}$ has nonempty intersection with $[0, 1]$ is fixed, and for each such k the roles of ψ and f in the previous argument can be exchanged to yield the estimate

$$|\langle f, \psi_{j,k}\rangle| \leq \|f\|_1 \|\psi\|_\infty 2^{j/2} \quad .$$

Thus, the two cases considered so far yield for each $\pm j = m \geq 0$ a constant number of coefficients of size $\leq C2^{-m/2}$. Sorting these coefficients by size yields a decay behaviour of order $O(2^{-m/2})$.

We next consider the remaining coefficients, i.e. $j \geq 0$ and $\mathrm{supp}(\psi_{j,k})$ does not contain a singularity. Here we use the vanishing moment condition, which requires for $0 \leq i < n$ that

$$\int_{\mathbb{R}} \psi(x)x^i dx = 0 \quad . \tag{1.2}$$

Note that this property is inherited by the $\psi_{j,k}$. Next we approximate f by its n-term Taylor polynomial of f at $x_0 = 2^{-j}k$, yielding

$$f(x) = \sum_{j=0}^{n-1} \frac{f^{(j)}(x_0)}{j!}(x - x_0)^j + R(x)$$

with $|R(x)| \leq C|x - x_0|^n \leq C2^{-jn}$, for a suitable constant C independent of k. Now (1.2) implies that

$$\begin{aligned} |\langle f, \psi_{j,k}\rangle| &= |\langle R, \psi_{j,k}\rangle| \\ &= \left| \int_{\mathbb{R}} R(x)\overline{\psi_{j,k}(x)}dx \right| \\ &\leq C2^{-jn}\|\psi_{j,k}\|_1 = C2^{-j(n+1/2)} \quad . \end{aligned}$$

Observe that there are $O(2^j)$ such coefficients. Hence, sorting these coefficients we arrive at a decay behaviour of $O(m^{-n-1/2})$. Thus, if we arrange the coefficients corresponding to both cases into a single sequence $(\theta_m)_{m\in\mathbb{N}}$ of decreasing modulus, we again obtain a decay of $O(m^{-n-1/2})$, which can be substituted into (1.1) to yield $\epsilon_N(f) \leq CN^{-2n}$.

The constructions presented in this chapter are to a large extent motivated by the desire to achieve a similar behaviour in two dimensions. First, however, we need to define the analogue of piecewise C^n. Our image domain is the square $[0,1[^2$. We call an image $f : [0,1[^2 \to \mathbb{R}$ *piecewise smooth* if it is of the form

$$f(x) = f_1(x) + \mathbf{1}_\Omega(x)f_2(x) \quad , \tag{1.3}$$

see Figure 1.2 for an illustration. Here $\mathbf{1}_\Omega$ is the indicator function of a compact subset $\Omega \subset [0,1[^2$ with a boundary $\partial\Omega$ that is C^2, by which we mean that there is a twice continuously differentiable parametrisation of $\partial\Omega$. The functions f_1 and f_2 belong to suitable classes of smooth functions, that may depend on the setting.

For the case that both f_1 and f_2 are C^2 as well, there exist theoretical estimates which yield that generally the optimal approximation rate will be of $O(N^{-2})$ [8]. We are going to show that wavelet bases fall short of this. For the following discussion, it suffices to assume that f_1 and f_2 are in fact constant. Observe that the estimates given below can be verified directly for the two-dimensional Haar wavelet basis and the special case that $f_1 = 1$, $f_2 = 0$, and Ω is the subset of $[0,1]$ below the diagonal $x = y$. This is a particularly simple example, where the pieces f_1 and f_2 are C^∞, the boundary is a straight line (hence C^∞) and not particularly ill-adapted to the tensor product setting, *and yet* wavelet bases show poor nonlinear

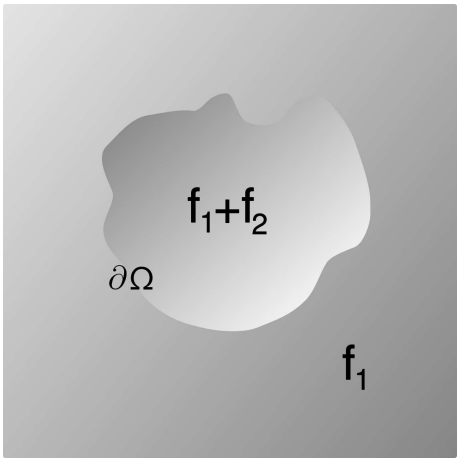

Figure 1.2: Example of a piecewise smooth function of the type (1.3).

approximation rates. As we will see below, the reason is that as in one dimension the wavelets situated at the boundary still contribute most to the signal energy, but this time their number is no longer under control.

We fix a two-dimensional wavelet basis, constructed in the usual way from a one-dimensional multiresolution analysis, see [23] or Chapter 1 of this volume. Our aim is to describe the approximation error of f in this basis. The argument follows the same pattern as for the one-dimensional case. Whenever a wavelet does not meet the boundary of Ω, the smoothness of the functions f_1, f_2 entails that the wavelet coefficients can be estimated properly. The problems arise when we consider those wavelets that meet the boundary. As before, for each wavelet of scale 2^{-j} meeting the boundary of Ω,

$$|\langle f, \psi_{j,k,l}\rangle| \sim 2^{-j} \quad .$$

Note that the tilde notation is a sharpened formulation, meaning that there exist upper *and* lower estimates between the two sides, at least for sufficiently many coefficients. (This is easily seen for the example involving Haar wavelets and diagonal boundary.) Hence, even though the scale-dependent decay behaviour for the coefficients corresponding to singularities is better than in one dimension, it holds for a crucially *larger* number of coefficients, which spoils the overall performance of the wavelet system. More precisely, as the supports of the wavelets are (roughly) squares of size $\sim 2^{-j}$ shifted along the grid $2^{-j}\mathbb{Z}^2$, the number of wavelets at scale 2^{-j} meeting the boundary is of $O(2^j)$. Thus we obtain $|\theta_m| \sim m^{-1}$, and this results in $\epsilon_N(f) \sim N^{-1}$.

A few observations are in order here: First, note that the arguments we present are indifferent to the smoothness of the boundary; for any smooth boundary of finite length that is not strictly horizontal or vertical we would obtain a similar behaviour. This is the blindness of wavelet tensor products to edge smoothness, that we already alluded to above: By construction, wavelets are only designed to represent discontinuities in the horizontal or vertical directions, and cannot be expected to detect connections between neighboring edge points. It should also be noted that the problem cannot be helped by increasing the number of vanishing moments of the wavelet system. (Again, this is readily verified for the diagonal boundary case.)

In the following subsections, we describe recently developed schemes that were designed to improve on this, at least for the continuous setting. The digitisation of these techniques will be the subject of Sections 1.2 and 1.3. The following remark contains a disclaimer that we feel to be necessary in connection with the transferal of notions and results from the continuous to the discrete domain.

Remark 1.1.1 *In this chapter we describe schemes that were originally designed for continuous image domains, together with certain techniques for digitisation of these notions. In this respect, this chapter reflects the current state of discussion. The continuous domain setting is in many ways more accessible to mathematical analysis, as witnessed by the smoothness properties that were at the basis of our discussion. The continuous domain viewpoint is advocated e.g. in the survey paper [15]; we specifically refer to [15, Section IV] for a discussion of the relevance of continuous domain arguments to coding.*

It is however not at all trivial to decide how results concerning asymptotic behaviour for continuous image domains actually apply to the analysis and design of image approximation schemes for the discrete setting. Observe that all nonlinear approximation results describing the asymptotic behavior for images with bounded domain necessarily deal with small scale *limits; for pixelised images, this limit is clearly irrelevant. Also, as we will encounter below, in particular in connection with the notion of angular resolution, the discussion of the continuous setting may lead to heuristics that hardly make sense for digital images.*

Note that also in connection with coding applications, the relevance of asymptotic results is not altogether clear: These results describe the right end of the nonlinear approximation curves. Thus they describe how effectively the approximation scheme adds finer and finer details, for numbers of coefficients that are already large, which in compression language means high bit rate coding. *By contrast, from the point of view of compression the left end of the nonlinear approximation curve is by far more interesting. As the approximation results in subsection 1.4.1 show, this is also where*

the new schemes show improved performance, somewhat contrary to the asymptotic results developed for the continuous setting.

1.1.1 Wedgelets

Wedgelets were proposed by Donoho [16] as a means of approximating piecewise constant images with smooth boundaries. The wedgelet dictionary by definition is given by the characteristic functions of wedge-shaped sets obtained by splitting dyadic squares along straight lines. It is highly redundant, and thus the problem of choosing a suitable representation or approximation of a signal arises. However, wedgelet approximation is not performed by pursuit algorithms or similar techniques typically encountered in connection with redundant dictionaries, but rather driven by a certain functional that depends on a regularisation parameter. As we will see below, this approach results in fast approximation algorithms.

For the description of wedgelets, let us first define the set of dyadic squares of size 2^{-j},

$$\mathcal{Q}_j = \{[2^{-j}k : 2^{-j}(k+1)[\times[2^{-j}\ell : 2^{-j}(\ell+1)[: 0 \leq k, \ell < 2^j\} \quad ,$$

and $\mathcal{Q} = \bigcup_{j=0}^{\infty} \mathcal{Q}_j$. A **dyadic partition** of the image domain is given by any partition (tiling) Q of $[0,1[^2$ into *disjoint* dyadic squares, not necessarily of constant size. A **wedgelet partition** is obtained by splitting each element $q \in Q$ of a dyadic partition Q into (at most) two *wedges*, $q = w_1 \cup w_2$, along a suitable straight line. The admissible lines used for splitting elements of $\mathcal{Q}_j$ are restricted to belong to certain prescribed sets L_j; we will comment on the choice of these sets below. A **wedgelet segmentation** is a pair (g, W) consisting of a wedge partition W, and a function g that is constant on all $w \in W$. See Figure 1.3 for an example.

A **wedgelet approximation** of an image f is now given as the minimiser of the functional

$$H_{\lambda,f}(g, W) = \|f - g\|_2^2 + \lambda |W| \quad , \tag{1.4}$$

over all admissible wedgelet segmentations (g, W). Here λ acts as a *regularisation* or *scale* parameter: For $\lambda = 0$, the minimisation algorithm will return the data f, whereas $\lambda \to \infty$ will eventually produce a constant image as minimiser. We denote the minimiser of $H_{\lambda,f}$ as $(\widehat{g}_\lambda, \widehat{W}_\lambda)$. The following remark collects the key properties that motivate the choice of wedgelets and the associated functional.

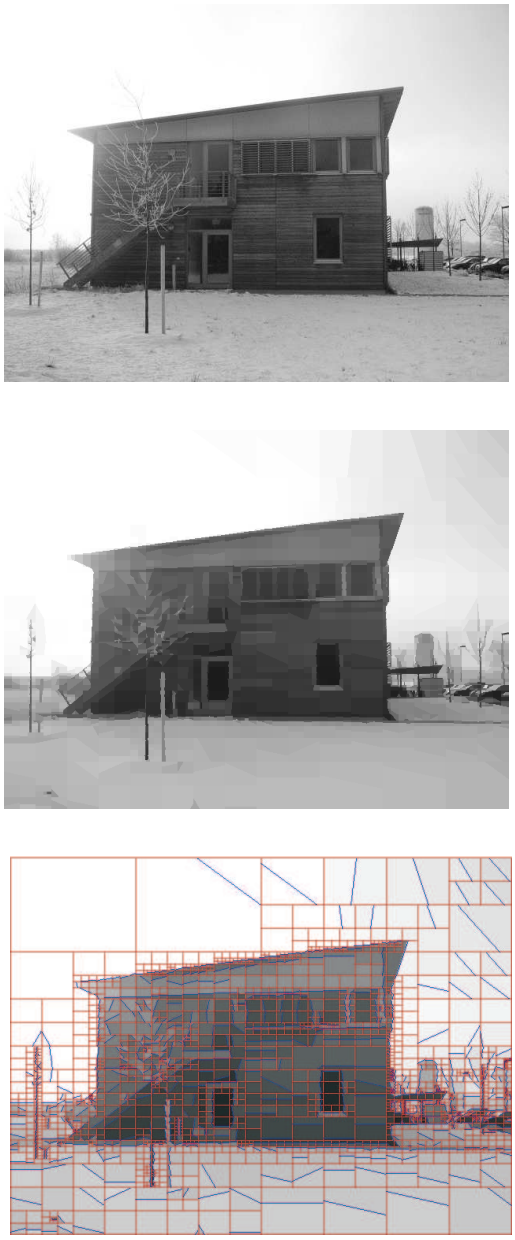

Figure 1.3: Image IBB North. (a) Original image, (b) wedge reconstruction $\lambda = 0.012$, and (c) with corresponding wedge grid superimposed.

Remark 1.1.2 (1) *Given the optimal partition $\widehat{W}_\lambda$, the optimal $\widehat{g}_\lambda$ is found by a simple projection procedure: For each $w \in \widehat{W}_\lambda$, $\widehat{g}_\lambda|_w$ is simply the mean value of g over w. Hence finding the optimal wedgelet segmentation is equivalent to finding the optimal partition.*
(2) *Dyadic partitions are naturally related to quadtrees. More precisely, given a dyadic partition W, consider the set V of all dyadic squares q such that there exists $p \in W$ with $p \subset q$. The inclusion relation induces a* quadtree structure *on V, i.e., V is a directed graph with the property that each node has either zero or four descendants. In this tree, W is just the set of* leaves, *i.e. the set of nodes without descendants.*

The quadtree structure is the basis for a fast algorithm for the computation of the optimal wedgelet segmentation W_λ, by recursive application of the following principle: Let $[0,1[^2 = q_1 \cup q_2 \cup q_3 \cup q_4$ be the decomposition into the four smaller dyadic squares. Then, for a fixed parameter λ, three cases may occur:

1. $\widehat{W}_\lambda = \{[0,1[^2\}$;
2. $\widehat{W}_\lambda$ *is obtained by a wedgesplit applied to* $[0,1[^2$;
3. $\widehat{W}_\lambda = \bigcup_{i=1}^4 \widehat{V}_\lambda^i$, *where each $\widehat{V}_\lambda^i$ is the optimal wedgelet segmentation of q_i associated to the restriction of f to q_i, and to the regularisation parameter λ.*

Note that for a fixed λ with minimiser $(\widehat{g}_\lambda, \widehat{W}_\lambda)$, $\widehat{g}_\lambda$ is the minimiser of the norm distance $\|f-g\|_2^2$ among all admissible wedgelet segmentations (g, W) with at most $N = |\widehat{W}_\lambda|$ wedges. This observation is used as the basis for the computation of nonlinear approximation rates below.

Let us next consider the nonlinear approximation behaviour of the scheme. The following technical lemma counts the dyadic squares meeting the boundary $\partial\Omega$. Somewhat surprisingly, the induced dyadic partition grows at a comparable speed.

Lemma 1.1.3 *Let f be piecewise constant, with C^2 boundary $\partial\Omega$. Let $\mathcal{Q}_j(f)$ denote the set of dyadic square $q \in \mathcal{Q}_j$ meeting $\partial\Omega$. Then, there exists a constant C such that $|\mathcal{Q}_j(f)| \leq 2^j C$ holds for all $j \geq 1$. Moreover, for each j there exists a dyadic partition W_j of $[0,1[^2$ containing $\mathcal{Q}_j(f)$, with $|W_j| \leq 3C2^j$.*

Proof. The statement concerning $\mathcal{Q}_j(f)$ is straightforward from a Taylor approximation of the boundary. The dyadic partition W_j is obtained inductively: W_{j+1} is obtained by replacing each dyadic square in $\mathcal{Q}_j(f)$ by the four dyadic squares of the next scale. Hence

$$|W_{j+1}| = |W_j| - |\mathcal{Q}_j(f)| + 4|\mathcal{Q}_j(f)| = |W_j| + 3|\mathcal{Q}_j(f)| \quad .$$

Now an easy induction shows the claim on $|W_j|$. □

We obtain the following approximation result. The statement is in spirit quite close to the results in [16], except that we use a different notion of resolution for the wedgelets, which is closer to our treatment of the digital case later on.

Theorem 1.1.4 *Let f be piecewise constant with C^2 boundary. Assume that the set L_j consists of all lines taking the angles $\{-\pi/2 + 2^{-j}\ell\pi : 0 \leq \ell < 2^j\}$. Then the nonlinear wedgelet approximation rate for f is $O(N^{-2})$, meaning that for $N \in \mathbb{N}$ there exists a wedgelet segmentation (g, W) with $|W| \leq N$ and $\|f - g\|_2^2 \leq CN^{-2}$.*

Proof. For $N = 2^j$, the previous lemma provides a dyadic partition W_j into $O(N)$ dyadic squares, such that $\partial\Omega$ is covered by the elements of $\mathcal{Q}_j \cap W_j$. Observe that only those dyadic squares contribute to the squared approximation error. In each such square, a Taylor approximation argument shows that the boundary can be approximated by a straight line in $O(2^{-2j})$ precision. The required angular resolution allows to approximate the optimal straight line by a line from L_j up to the same order of precision. Now the incurred squared L^2-error is of order $O(2^{-3j})$; the additional $O(2^{-j})$ factor is the diameter of the dyadic squares. Summing over the $O(2^j)$ squares yields the result. □

We note that the theorem requires that the number of angles increases as the scale goes to zero; the *angular resolution of L_j* scales linearly with 2^j. Observe that this requirement does not make much sense as we move on to digital images. In fact, this is the first instance where we encounter the phenomenon that intuitions from the continuous model prove to be misleading in the discrete domain.

1.1.2 Curvelets

Curvelets are in many ways conceptually closer to wavelets than wedgelets. While wedgelet approximation may be read as the approximation of the image using building blocks that are specifically chosen for the image at hand, curvelet approximation is based on the decomposition of the image into a *fixed* system of components, prescribed without prior knowledge of the image. The **curvelet system** is a family of functions $\gamma_{j,l,k}$ indexed by a scale parameter j, an orientation parameter l and a position parameter $k \in \mathbb{R}^2$, yielding a **normalised tight frame** of the image space. The latter property amounts to postulating for all $f \in \mathrm{L}^2(\mathbb{R}^2)$

$$\|f\|_2^2 = \sum_{j,k,l} |\langle f, \gamma_{j,k,l}\rangle|^2 \quad ,$$

or equivalently, that each f can be expanded as

$$f = \sum_{j,k,l} \langle f, \gamma_{j,k,l} \rangle \gamma_{j,k,l} \quad ,$$

see [23, Chapter V]. This expansion is reminiscent of orthonormal bases, however, the elements of a tight frame need not be pairwise orthogonal; in fact they need not even be linearly independent.

In the curvelet setting, image approximation is performed by expanding the input in the curvelet frame and quantizing the coefficients, just as in the wavelet setting. However, the effectiveness of the approximation scheme critically depends on the *type of scaling*, and the sampling of the various parameters. Unlike the classical, group theoretical construction of 2D wavelets, as introduced by Antoine and Murenzi [1, 2], which also incorporates scale, orientation and position parameters, the scaling used in the construction of curvelets is *anisotropic*, resulting in atoms that are increasingly more needle-like in shape as the scale decreases.

Let us know delve into the definition of curvelets. The following construction is taken from [6], which describes the most recent generation of curvelets. A precursor was constructed in [5] ("curvelets 99" in the terminology of [6]), which has a more complicated structure, relying on additional windowing and the ridgelet transform. A comparison of the two types of curvelets is contained in [6]. Both constructions are different realisations of a core idea which may be summarised by the catchphrase that the curvelet system corresponds to a *critically sampled, multiscale directional filterbank, with angular resolution behaving like* $1/\sqrt{scale}$.

As the filterbank view suggests, curvelets are most conveniently constructed on the frequency side. The basic idea is to cut the frequency plane into subsets that are cylinders in polar coordinates. The cutting needs to be done in a smooth way however, in order to ensure that the resulting curvelets are rapidly decreasing.

For this purpose, we fix two window functions,

$$\nu : [-\pi, \pi] \to \mathbb{C} \quad , \quad w : \mathbb{R}^+ \to \mathbb{C} \quad .$$

Both ν and w are assumed smooth; in addition, for ν we require that its 2π-periodic extension ν_{per} is smooth as well. In addition, we pick w to be compactly supported. ν acts as angular window; in order to guarantee that the functions constructed from ν are even. Moreover, we impose that ν fulfills (for almost every ϑ)

$$|\nu_{per}(\theta)|^2 + |\nu_{per}(\theta + \pi)|^2 = 1 \quad , \tag{1.5}$$

which guarantees that the design of curvelets later on covers the full range of angles. Equation (1.5) allows to construct partitions of unity of the

angular domain into a dyadic scale of elements: Defining $\nu_{j,l} = \nu(2^j\theta - \pi l)$, for $l = 0, \ldots, 2^j - 1$ on $[-\pi, \pi]$, and extended periodically, it is easily verified that

$$\sum_{l=0}^{2^j-1} |\nu_{j,l}(\theta)|^2 + |\nu_{j,l}(\theta+\pi)|^2 = 1 \quad . \tag{1.6}$$

A similar decomposition is needed for the scale variable. Here we make the additional assumption that

$$|w_0(s)|^2 + \sum_{j=1}^{\infty} |w(2^j s)|^2 = 1 \quad , \tag{1.7}$$

for a suitable compactly supported C^∞-function w_0. w should be thought of as a bump function concentrated in the interval $[1, 2]$.

In the following we will frequently appeal to polar coordinates, i.e., we will identify $\xi \in \mathbb{R}^2 \setminus \{0\}$ with $(|\xi|, \theta_\xi) \in \mathbb{R}^+ \times]-\pi, \pi]$. The scale and angle windows allow a convenient control on the design of the curvelet system. The missing steps are now to exert this control to achieve the correct scale-dependent angular resolution, and to find the correct sampling grids (which shall depend on scale and orientation). For the first part, filtering with the scale windows $(w_j)_{j\geq 0}$ splits a given signal into its frequency components corresponding to the annuli $A_j = \{\xi \in \mathbb{R}^2 : 2^j \leq |\xi| < 2^{j+1}\}$. In each such annulus, the number of angles should be of order $2^{j/2}$, following the slogan that the angular resolution should scale as $1/\sqrt{scale}$. Thus we define the scale-angle window function $\eta_{j,l}$, for $j \geq 1$ and $0 \leq l \leq 2^{\lfloor j/2 \rfloor} - 1$, on the Fourier side by

$$\widehat{\eta}_{j,l}(\xi) = w_j(|\xi|)(\nu_{2^{\lfloor j/2 \rfloor},l}(\theta_\xi) + \nu_{2^{\lfloor j/2 \rfloor},l}(\theta_\xi + \pi)) \quad . \tag{1.8}$$

In addition, we let $\widehat{\eta}_{0,0}(\xi) = w_0(|\xi|)$, which is responsible for collecting the low-frequency part of the image information. Up to normalisation and introduction of the translation parameter, the family $(\eta_{j,l})_{j,l}$ is the curvelet system. By construction, $\widehat{\eta}_{j,l}$ is a function that is concentrated in the two opposite wedges of frequencies

$$W_{j,l} = \{\xi \in \mathbb{R}^2 : 2^j \leq |\xi| \leq 2^{j+1} \ , \ \theta_\xi \text{ or } \theta_\xi + \pi \in [2^{-\lfloor j/2 \rfloor} l, 2^{-\lfloor j/2 \rfloor}(l+1)]\} \ ,$$

as is illustrated in Figure 1.4.

Now (1.7) and (1.6) implies for almost every $\xi \in \mathbb{R}^2$ that

$$\sum_{j,l} |\widehat{\eta}_{j,l}(\xi)|^2 = 1 \quad , \tag{1.9}$$

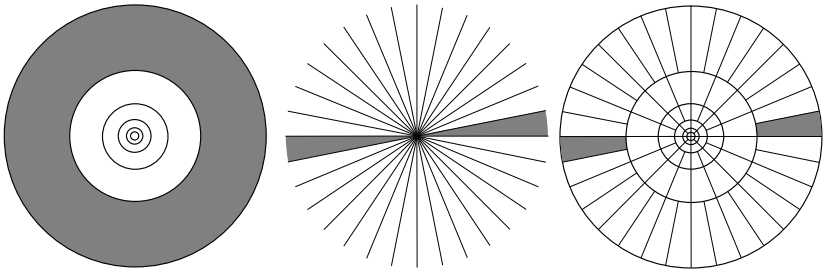

Figure 1.4: Idealised frequency response of the curvelet system. Filtering using the scale window w_j, followed by filtering with the angular window $\nu_{j,l}$, yields the frequency localisation inside the wedge $W_{j,l}$. Observe the scaling of the angular resolution, which doubles at every other dyadic step.

and standard arguments allow to conclude from this that convolution with the family $(\eta_{j,l})_{j,l}$ conserves the L^2-norm of the image, i.e.

$$\|f\|_2^2 = \sum_{j,l} \|f * \eta_{j,l}\|_2^2 \quad . \tag{1.10}$$

The definition of the **curvelet frame** is now obtained by critically sampling the isometric operator $f \mapsto (f * \eta_{j,l})_{j,l}$. The following theorem states the properties of the resulting system, see [6] for a proof.

Theorem 1.1.5 *There exist frequency and scale windows ν, w, normalisation constants $c_{j,l} > 0$ and sampling grids $\Gamma_{j,l} \subset \mathbb{R}^2$ (for $j \geq 0, l = 0, \ldots, 2^{\lfloor j/2 \rfloor} - 1$) with the following properties: Define the index set*

$$\Lambda = \{(j,l,k) : j \geq 0, l = 0, \ldots, 2^{\lfloor j/2 \rfloor}, k \in \Gamma_{j,l}\} \quad .$$

Then the family $(\gamma_\lambda)_{\lambda \in \Lambda}$, defined by

$$\gamma_{j,l,k}(x) = c_{j,l} \eta_{j,l}(x - k)$$

is a normalised tight frame of $\mathrm{L}^2(\mathbb{R})$, yielding an expansion

$$f = \sum_{\lambda \in \Lambda} \langle f, \gamma_\lambda \rangle \gamma_\lambda \ , \quad \forall f \in \mathrm{L}^2(\mathbb{R}^2). \tag{1.11}$$

The following list collects some of the geometric features of the curvelet system:

1. The shift in the rotation parameter implies that $\gamma_{j,l}(x) = \gamma_{j,0}(R_{\theta_{j,l}} x)$, where $\theta_{j,l} = \pi l 2^{-\lfloor j/2 \rfloor}$, and R_θ denotes the rotation matrix

$$R_\theta = \begin{pmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{pmatrix} \quad .$$

2. $\widehat{\gamma}_{j,0,0}$ is essentially supported in a union of two rectangles of dimensions $O(2^j \times 2^{\lfloor j/2 \rfloor})$, and the associated sampling lattice can be chosen as $\Gamma_{j,0} = \delta_{1,j}\mathbb{Z} \times \delta_{2,j}\mathbb{Z}$, with $\delta_{1,j} \sim 2^{-j}$ and $\delta_{2,j} \sim 2^{-\lfloor j/2 \rfloor}$. As could be expected from the previous observation, $\Gamma_{j,l} = R_{\theta_{j,l}}\Gamma_{j,0}$.

3. By the previous observation, the change of sampling lattices from j to $j+2$ follows an *anisotropic scaling law*,

$$\Gamma_{j+2,0} \approx D^{-1}\Gamma_{j,0} \ , \quad \text{where } D = \begin{pmatrix} 4 & 0 \\ 0 & 2 \end{pmatrix}$$

   The discussion in [6] suggests that, at least conceptually, it is useful to think of all curvelets to be descended from the two basic curvelets $\gamma_{1,0}$ and $\gamma_{2,0}$, by iterating the relation $\gamma_{j+2,0}(x) \approx \det(D)^{1/2}\gamma_{j,0}(Dx)$.

4. Summarizing, the $\gamma_{j,l}$ are a system of rapidly decreasing functions that oscillate at speed of order $2^{\lfloor j/2 \rfloor}$, primarily in the $(\cos(\theta_{j,l}), \sin(\theta_{j,l}))$ direction. As $j \to \infty$, the essential support of $\gamma_{j,l,0}$ scales like a rectangle of size $2^{-j} \times 2^{-\lfloor j/2 \rfloor}$, when viewed in the appropriate coordinate system.

The following theorem shows that up to a logarithmic factor the curvelet system yields the desired nonlinear approximation behaviour for piecewise C^2 functions. One of the remarkable features of the theorem is that the approximation rate is already achieved by simple nonlinear truncation of (1.11). Observe that this is identical with best N-term approximation only for orthogonal bases; however, the curvelet system is only a tight frame, and cannot be expected to be an orthogonal basis.

Theorem 1.1.6 ([6, Theorem 1.3])
Let f be a piecewise C^2 function with C^2 boundary. For $N \in \mathbb{N}$, let $\Lambda_N(f) \subset \Lambda$ denote the indices of the N largest coefficients. Then there exists a constant $C > 0$ such that

$$\|f - \sum_{\lambda \in \Lambda_N} \langle f, \gamma_\lambda \rangle \gamma_\lambda\|_2^2 \leq C N^{-2} (\log N)^3 \quad . \tag{1.12}$$

For a detailed proof of the theorem we refer to [6]. In the following we present a shortened version of the heuristics given in [6]. They contrast nicely to the wavelet case discussed above, and motivate in particular the role of anisotropic scaling for the success of the curvelet scheme. A graphically intuitive way of understanding the proof is to first observe that the locally smooth parts of the image are captured largely by the translates

of the low-pass filter $\gamma_{0,0,0}$. The remaining image information is therefore located in the edge, and the whole point of the curvelet construction is to provide "brushstrokes" that are increasingly precise as their scale goes to zero, both in the possible orientations and in their breadth (due to the anisotropic scaling).

Suppose we are given a piecewise smooth image f, and a fixed scale index j. We start the argument by a geometric observation that motivates the use of anisotropic scaling and rotation: Recall from the wavelet case that $O(2^j)$ dyadic squares of size 2^{-j} are needed to cover the boundary $\partial\Omega$. This time we consider a covering by rectangles of size $2^{-\lfloor j/2 \rfloor} \times 2^{-j}$, which may be arbitrarily rotated. Then a Taylor approximation of the boundary shows that this can be done by $O(2^{j/2})$ such rectangles, which is a vital improvement over the wavelet case.

Next we want to obtain estimates for the scalar products $\langle f, \gamma_{j,l,k} \rangle$, depending on the position of the curvelet relative to the boundary. Recall that $\gamma_{j,l,k}$ is a function that has elongated essential support of size $2^{-\lfloor j/2 \rfloor} \times 2^{-j}$, in the appropriately rotated coordinate system, and oscillates in the "short" direction.

Then there are basically three cases to consider, sketched in Figure 1.5:

1. *Tangential:* The essential support of $\gamma_{j,l,k}$ is close in position and orientation to one of the covering boxes, i.e., it is tangent to the boundary.

2. *Transversal:* The essential support is close in position to one of the covering boxes, but not in orientation. Put differently, the support intersects the boundary at a significant angle.

3. *Disjoint:* The essential support does not intersect the boundary.

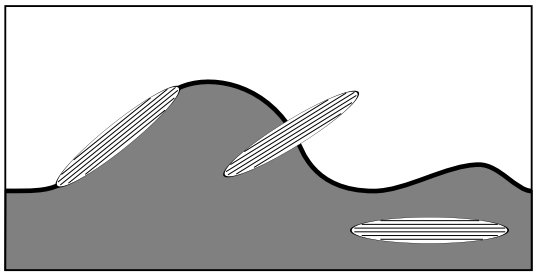

Figure 1.5: An illustration of the three types of curvelet coefficients. The essential supports of the curvelets are shown as ellipses, with indicated oscillatory behaviour. From left to right tangential, transversal and disjoint case.

The argument rests on the intuition that only the tangential case yields significant coefficients. One readily expects that the disjoint case leads to negligible coefficients: the image is smooth away from the boundary, hence the oscillatory behavior of the curvelet will cause the scalar product to be small. By looking more closely at the *direction* of the oscillation, we can furthermore convince ourselves that the transversal case produces negligibly small coefficients as well: The predominant part of the essential support is contained in regions where f is smooth, and the oscillations across the short direction imply that this part contributes very little to the scalar product.

Thus we have successfully convinced ourselves that only the tangential case contributes significant coefficients. Here we apply the same type of estimate that we already used in the wavelet cases, i.e.

$$|\langle f, \gamma_{j,l,k}\rangle| \leq \|f\|_\infty \|\gamma_{j,l,k}\|_1 \leq C 2^{-3j/4} \quad ,$$

due to the choice of normalisation coefficients $c_{j,l}$. Since there are $O(2^{j/2})$ boxes, the sampling of the position and angle parameter in the curvelet system implies also $O(2^{j/2})$ coefficients belonging to the tangential case. Ignoring the other coefficients, we therefore have produced –rather informal– evidence for the statement that the sorted coefficients obey the estimate

$$|\theta_m| \leq C m^{-3/2} \quad .$$

As we have already mentioned above, the normalised tight frame property of the curvelet system is equivalent to the statement that the *coefficient operator* assigning each function f its curvelet coefficients $(\langle f, \gamma_{j,l,k}\rangle_{(j,l,k)\in\Lambda})$ is an isometric mapping from $\mathrm{L}^2(\mathbb{R}^2)$ to $\ell^2(\Lambda)$, the latter being the Hilbert space of square-summable coefficient sequences. The adjoint of this operator is the *reconstruction operator* mapping each coefficient sequence $(\alpha_\lambda)_{\lambda\in\Lambda} \in \ell^2(\Lambda)$ to the sum $\sum_{\lambda\in\Lambda} \alpha_\lambda \gamma_\lambda$; as the adjoint of an isometry, this operator is normdecreasing. This allows to finish the argument by the estimate

$$\begin{aligned} \|f - \sum_{\lambda\in\Lambda_N} \langle f, \gamma_\lambda\rangle \gamma_\lambda\|_2^2 &= \|\sum_{\lambda\in\Lambda\setminus\Lambda_N} \langle f, \gamma_\lambda\rangle \gamma_\lambda\|_2^2 \leq \sum_{\lambda\in\Lambda_N} |\langle f, \gamma_\lambda|^2 \\ &= \sum_{m=N+1}^{\infty} |\theta_m|^2 \leq C N^{-2} \quad . \end{aligned}$$

Observe that the logarithmic factor in the statement of Theorem 1.1.6 has disappeared in the course of the argument. This is just an indicator of the degree of oversimplification of the presentation.

Remark 1.1.7 *We close the section by citing another observation from [6], which allows a neat classification of wavelet, curvelet and ridgelet schemes by means of their angular resolution: Wavelets have a constant angular resolution, for curvelets the angular resolution behaves like* $1/\sqrt{\text{scale}}$*, for ridgelets like* $1/\text{scale}$*.*

1.1.3 Alternative approaches

Beside the two methods described in the previous subsections, various recent models for the information content of natural images were developed from different heuristic principles. It is outside the scope of this chapter to describe all of them in detail; in the following we briefly sketch some of the more prominent approaches.

Another interesting recently developed method that combines wavelet theory with geometric partitioning of the image domain are bandelets [20, 21], which make use of redundancies in the geometric flow, corresponding to local directions of the image grey levels considered as a planar onedimensional field. The geometry of the image is summarised with local clustering of similar geometric vectors, the homogeneous areas being taken from a quadtree structure. A bandelet basis can be viewed as an adaptive wavelet basis, warped according to the locally selected direction. Bandelet decomposition achieves optimal approximation rates for C^α functions. This method presents similarities with optimal wedgelet decompositions in that it uses geometric partitioning of the image domain, according to the minimisation of a certain complexity-distortion functional. For instance, bandelets decomposition combined with a rate-distortion method, leads to a quite competitive compression scheme. The geometric part can be used to incorporate prior knowledge, allowing to adapt the scheme to very specific classes of images, such as ID photos; see the website [22] for the description of a bandelet coder of ID images.

Another approach that uses oriented wavelet bases has been introduced under the name "directionlets" [30]. Like wavelets, directionlets rely on separable filters, however, the filtering is performed along digital lines that are not necessarily horizontal or vertical, resulting in improved approximation rates for certain classes of primarily geometric images.

A further approximation scheme is based on the use of triangulations, which corresponds to a quite different philosophy. By their flexibility, adaptive irregular triangulations have very good approximation behavior. It can be shown that the optimal rates of approximation can be attained (see [21]) when we require that every conform triangulation is allowed for the representation. The main problem encountered by these methods is the sheer number of possible triangulations. In practice, especially for the purpose

of implementation, one is forced to consider highly reduced triangulations classes, while still trying to obtain nearly optimal results.

To mention an example of such an approach, the method proposed in [10] uses a greedy removal of pixels, minimizing at each step the error among the possible triangulations. The class of triangulations under consideration is reduced to the set of Delaunay triangulations of a finite set of pixel positions, which allows a simplified parameterisation, only using the point coordinates, without any connectivity information about the according triangulation. This fact is employed in a suited scattered data compression scheme. For natural images, the rate-distortion performances achieved are comparable with those obtained by wavelet methods, leading to very different kind of artefacts. In particular it avoids ringing artefacts, but smooths textured areas.

An approach which is in a sense dual to the majority of the schemes described here are *brushlets*, introduced by Meyer and Coifman [24]. While most of the approaches we mentioned so far involve some form of spatial adaptation to the image content, brushlet approximations are based on the adaptive tiling of the frequency plane. As might be expected from this description, the experiments in [24] show that brushlets are quite well adapted to the representation of periodic textures, which shows that the brushlet approach is in a sense complementary to geometric approaches such as wedgelets. By construction, brushlets have trouble dealing with piecewise smooth images, which constitute the chief class of benchmark signals in this chapter. It is well-known that the Fourier transform is particularly ill-suited to dealing with piecewise smooth data. Hence any scheme that uses the Fourier transform of the image as primary source of information will encounter similar problems; for brushlets this effect can also be examined in the examples displayed in [24].

Finally, let us mention dictionary-based methods, usually employed in connection with pursuit algorithms. As most of the approaches described in this chapter are based more or less explicitly on redundant systems of building blocks, there are necessarily some similarities to dictionary-based methods. The use of highly redundant dictionaries for image representations is the subject of a separate chapter in this volume, to which we refer the interested reader.

1.2 Digital Wedgelets

Let us now turn to discrete images and algorithms. In this section we describe a digital implementation of Donoho's wedgelet algorithm. For notational convenience, we suppose the image domain to be $\Omega = \{0, \dots, 2^J - 1\} \times \{0, \dots, 2^J - 1\}$. In this setting, dyadic squares are sets of the type

$\{2^j k, \dots, 2^j(k+1)-1\} \times \{2^j \ell, 2^j(\ell+1)-1\}$, with $0 \le k, \ell < 2^{J-j}$. Our goal is to describe an efficient algorithm that for a given image $f \in \mathbb{R}^\Omega$ computes a minimiser of

$$H_{\lambda,f}(g, W) = \|f - g\|_2^2 + \lambda |W| \quad , \tag{1.13}$$

where W is a wedge partition of Ω and g is constant on each element of W. As in the continuous setting, wedge partitions are obtained from dyadic partitions by splitting dyadic squares along straight lines. It turns out that there are several options of defining these notions; already the digitisation of lines is not as straightforward an issue as one might expect.

In the following, we use the definitions underlying a recent implementation, described in more detail in [11, 19]. Other digitisations of wedgelets can be used for the design of wedgelet algorithms, and to some extent, the following definitions are just included for the sake of concreteness. However, as we explain below, they also provide particularly fast algorithms.

We fix a finite set $\Theta \subset]-\pi/2, \pi/2]$ of *admissible angles*. The admissible discrete wedgelets are then obtained by splitting dyadic squares along lines meeting the x-axis at an angle $\theta \in \Theta$.

Definition 1.2.1 *For $\theta \in]-\pi/2, \pi/2]$ let $v_\theta^\perp = (-\sin(\theta), \cos(\theta))$. Moreover, define*

$$\delta = \max\{|\sin(\theta)|/2, |\cos(\theta)|/2\} \quad .$$

The **digital line through the origin in direction** *v_θ is then defined as*

$$L_{0,\theta} = \{p \in \mathbb{Z}^2 : -\delta < \langle p, v_\theta^\perp \rangle \le \delta\} \quad . \tag{1.14}$$

Moreover, we define $L_{n,\theta}$, for $n \in \mathbb{Z}$, as

$$L_{n,\theta} = \begin{cases} \{p + (n, 0) : p \in L_{0,\theta}\} & : \quad |\theta| > \pi/4 \\ \{p + (0, n) : p \in L_{0,\theta}\} & : \quad |\theta| \le \pi/4 \end{cases} \quad .$$

In other words, $L_{n,\theta}$ is obtained by shifting $L_{0,\theta}$ by integer values in the vertical direction for **flat lines**, and by shifts in the horizontal direction for **steep lines**. In [11] we prove that the set $(L_{n,\theta})_{n\in\mathbb{Z}}$ partitions $\mathbb{Z}^2$, i.e. $\mathbb{Z}^2 = \bigcup_{n\in\mathbb{Z}}^\bullet L_{n,\theta}$; see also [19, Section 3.2.2].

Now we define the **discrete wedge splitting** of a square.

Definition 1.2.2 *Let $q \subset \mathbb{Z}^2$ be a square, and $(n, \theta) \in \mathbb{Z} \times [-\pi/2, \pi/2[$. The* **wedge split induced by** *$L_{n,\theta}$ is the partition of q into the sets $\{w_{n,\theta}^1(q), w_{n,\theta}^2(q)\}$ defined by*

$$\begin{cases} w_{n,\theta}^1(q) = \bigcup\limits_{k \le n} L_{k,\theta} \cap q \\ w_{n,\theta}^2(q) = \bigcup\limits_{k > n} L_{k,\theta} \cap q \end{cases} \quad .$$

Our description of discrete wedgelets is somewhat nonstandard due to the fact that we use a globally defined set of angles and lines. The advantage of our definition is that it allows the efficient solution of the key problem arising in rapid wedgelet approximation, namely the efficient computation of image mean values over wedges of varying shapes and sizes. In a sense, this latter problem is the only serious challenge that is left after we have translated the observations made for the continuous setting in Remark 1.1.2 to the discrete case: Again the minimisation problem is reduced to finding the best wedgelet segmentation, and the recursive minimisation procedure is fast, provided that for every dyadic interval the optimal wedgesplit is already known. The latter problem requires the computation of mean values, for large numbers of wedges, and here our definitions pay off.

In the following two subsections we give algorithms dealing with a somewhat more general model, replacing locally constant by locally polynomial approximation. In other words, we consider minimisation (1.13) for functions g that are given by polynomials of fixed degree r on the elements of the segmentation. Thus the following also applies to the platelets introduced in [35]. The more general problem requires the computation of higher degree image moments over wedge domains, but is otherwise structurally quite similar to the original wedgelet problem. There exists a freely available implementation of models up to order two, which can be downloaded from [34].

Subsection 1.2.1 sketches the key technique for moment computation. It relies on precomputed lookup-tables containing cumulative sums over certain image domains. The number of lookup-tables grows linearly with the number of angles in Θ and the number of required moments. This way, the *angular resolution* of the discrete wedges can be prescribed in a direct and convenient way, and at linear cost, both computational and in terms of memory requirements. Subsection 1.2.2 contains a summary of the algorithm for the minimisation of (1.13). For more details we refer to [11, 19].

1.2.1 Rapid summation on wedge domains: Discrete Green's theorem

As explained above, efficient wedgelet approximation requires the fast evaluation integrals of the form $\int_w f(x,y)dxdy$, over all admissible wedges w. For higher order models, image moments of the form $\int_w f(x,y)x^i y^j dxdy$ need to be computed; for the discrete setting, the integral needs to be replaced by a sum. The following is a sketch of techniques developed to provide a fast solution to this problem.

For exposition purposes, let us first consider the continuous setup. We let $\mathcal{Q}_+$ denote the positive quadrant, $\mathcal{Q}_+ = \mathbb{R}_0^+ \times \mathbb{R}_0^+$. Given $z \in \mathcal{Q}_+$,

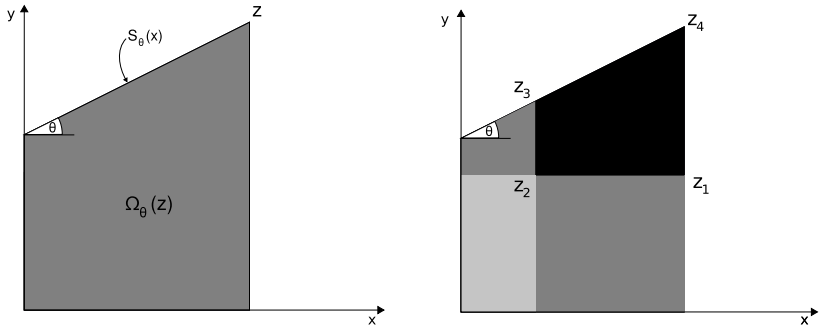

Figure 1.6: Left: The sets $S_\theta(z)$ and $\Omega_\theta(z)$. Right: An illustration of the argument proving (1.16).

and $\theta \in]-\pi/2, \pi/2]$, let $S_\theta(z) = z + \mathbb{R}^-(\cos(\theta), \sin(\theta)) \cap \mathcal{Q}^+$. Moreover, denote by $\Omega_\theta(z) \subset \mathcal{Q}^+$ the domain that is bounded by the coordinate axes, the vertical line through z, and $S_\theta(z)$, see Figure 1.6. Define the auxiliary function $K_\theta : \mathcal{Q}^+ \to \mathbb{R}$ as

$$K_\theta(z) = \int\limits_{\Omega_\theta(z)} f(x,y)\ dx\ dy \quad ; \tag{1.15}$$

note that this implies $K_{\pi/2} = 0$, as the integral over a set of measure zero.

Let us now consider a wedge of fixed shape, say a trapezoid w, with corners z_1, z_2, z_3, z_4, as shown in the right hand part of Figure 1.6. Then (1.15) implies that

$$\int\limits_w f(x,y)\ dx\ dy = K_\theta(z_4) - K_\theta(z_3) - K_0(z_1) + K_0(z_2). \tag{1.16}$$

In order to see this, observe that $w = \Omega_\theta(z_4) \setminus (\Omega_\theta(z_3) \cup \Omega_0(z_1))$. Hence the integral over w is obtained by subtracting from $K_\theta(z_4)$ the integrals over $\Omega_\theta(z_3)$ and $\Omega_0(z_1)$, and then adding the part that is subtracted twice, i.e. the integral over $\Omega_\theta(z_3) \cap \Omega_0(z_1) = \Omega_0(z_2)$.

Note that the evaluation of the right-hand side of (1.16) involves only 4 operations, *supposing that K_θ and K_0 are known.* Similar results can then be obtained for the different kind of wedge domains arising in the general scheme, and more generally for all polygonal domains with boundary segments belonging to angles in Θ. As a side remark, these considerations in fact just describe a special case of Green's theorem; see [11] for a more complete discussion of this connection, and for a description of the discrete implementation of this formula.

The discrete analogs K_θ^d of the auxiliary functions can be stored in matrices of the same size as the image, and they are efficiently computable in linear time, by cumulative summation first in the vertical direction, and then along the lines $L_{n,\theta}$. As the main result of this discussion, we record the following theorem. For the rest of this section, N denotes the number of pixels, $N = 2^{2J}$.

Theorem 1.2.3 *For any angle $\theta \in]-\pi/2, \pi/2]$, the auxiliary matrix K_θ^d is computable in $O(N)$. After computing K_θ^d and K_0^d, the sum $\sum_{(x,y)\in w} f(x,y)$ is obtainable using at most 6 operations, for every wedge domain w obtained by splitting a dyadic square along a line with angle θ.*

1.2.2 Implementation

Now, combining Donoho's observations from Remark 1.1.2 with the techniques outlined in the previous section, we obtain the following algorithm for the minimisation of (1.13).

1. Compute the auxiliary matrices $K_{\theta,i,j}^d$, for all $\theta \in \Theta$, which are necessary for the computation of the moment of index i, j to be used in the next steps. Local regression of order r requires $(r+1)(r+2)/2$ such moment matrices. By the considerations in the previous subsection, the overall memory and time requirements for this computation step is therefore $(r+1)(r+2)/2 \times N \times |\Theta|$.

2. For each dyadic square q, we need to select a best local wedge-regression model among the possible wedge splitting of this square. For each digital line l, we compute the $(r+1)(r+2)/2$ moments in fixed time, using the appropriate version of (1.16). This allows to solve the corresponding regression problems over w_l^1 and w_l^2, which requires $O(r^3)$ flops. Finally, we compute the according discrete l^2-error. This procedure applies to the $|\Theta|2^{j+1}$ admissible discrete lines passing through q.

   For each q, we need then to store the line $\hat{l}_{n_q,\theta_q}$ which corresponds to the minimal error, the associated two sets of optimal coefficients of the local regression models, and the incurred squared error E_q.

   The whole procedure needs to be carried out for all $\frac{2N-1}{3}$ dyadic squares.

3. Once Step 2. is finished, we are in a position to determine the wedgelet partition $\widehat{W_\lambda}(f)$ which minimises (1.13) for a given parameter λ, using the algorithm sketched in Remark 1.1.2. The algorithm

runs through all dyadic squares, starting from the smallest ones, i.e. single pixels.

Hence, if we consider a dyadic square q, its children q_i, $i = 1, \ldots, 4$ have already been treated, and we know an optimal partition for each q_i, denoted by $\hat{W}_\lambda(q_i)$, and also the associated error $E_{q_i,\lambda}$ and penalisation $\lambda|(\hat{W}_\lambda(q_i))|$.

The optimal partition of A is then the result of the comparison of two partitions, $\widehat{W}(q)$ and $\cup_{i=1}^4 \widehat{W}_\lambda(q_i)$. The associated error is given by

$$E_{q,\lambda} = \min\{E_q + \lambda|\hat{W}(q)|, \sum_{i=1}^{4} E_{q_i,\lambda}\} \quad , \tag{1.17}$$

and according to the result of the comparison, we store the corresponding optimal wedge partition $\widehat{W}_{q,\lambda}$. The process stops at the top level, yielding the minimiser $(\widehat{g}_\lambda, \widehat{W}_\lambda)$.

We summarise the results concerning the computational costs in the following proposition.

Proposition 1.2.4 *Step 1. requires $O(aNr^3)$ flops and a memory storage of $O(aN^2r^3)$. Step 2. also requires $O(aNr^3)$ flops and a memory storage in $O(rN)$. Step 3. requires $O(N)$ flops.*

The following observations are useful for fine-tuning the algorithm performance:

Remark 1.2.5 *(a) In actual implementation, allocation for the auxiliary matrices storing K_θ^d turns out to be a crucial factor. A closer inspection shows that in the steps 1. and 2., the angles in Θ can be treated* consecutively*, thus reducing memory requirement to $O(Nr^3)$. This results in a considerable speedup.*
(b) The use of a fixed set Θ of angles for splitting dyadic squares of varying size is not very efficient: For small dyadic squares, a small difference in angles yields identical wedgesplits. Roughly speaking, a dyadic square of size 2^j can resolve $O(2^j)$ angles. It is possible to adapt the algorithm to this scaling. Note that this scale-dependent angular resolution is precisely the inverse of what is prescribed by Theorem 1.1.4. Numerical experiments, documented in [19, Sect.6.3], show that this scale dependent angular resolution leads to the same approximation rates as the use of a full set of 2^J angles, valid for all scales.
(c) A further highly interesting property of the algorithm is the fact that only the last step uses the regularisation parameter. Thus the results of the previous steps can be recycled, allowing fast access to $(\widehat{g}_\lambda, \widehat{W}_\lambda)$ for arbitrary *parameters λ.*

1.3 Digital Curvelets: Contourlets

The curvelet construction relies on features that are hard to transfer to the discrete setting, such as polar coordinates and rotation. Several approaches to digital implementation have been developed since the first inception of curvelets, see e.g. [28, 12, 13]. In the following we present the approach introduced by Do and Vetterli [13], which to us seems to be the most promising among the currently available implementations, for several reasons: It is based on fast filterbank algorithms with perfect reconstruction; i.e., the tight frame property of curvelets is fully retained, in an algorithmically efficient manner. Moreover, the redundancy of the transform is 1.333, which is by far better than the factor $16J+1$ (J = number of dyadic scales in the decomposition) reported in [28]. It is clear that from a coding perspective, redundancy is a critical issue.

The starting point for the construction of contourlets is the observation that computing curvelet coefficients can be broken down into the following three steps, compare Figure 1.4:

1. Bandpass filtering using the scale windows w_j.
2. Directional filtering using the angular windows $\nu_{j,l}$.
3. Subsampling using the grids $\Gamma_{j,l}$, resulting in a tight frame expansion with associated inversion formula.

The discrete implementation follows an analogous structure, see Figure 1.7.

1. The image is passed through a pyramid filterbank, yielding a sequence of bandpassed and subsampled images.
2. Directional filterbanks [3, 12], are applied to the difference images in the pyramid, yielding directionally filtered and critically subsampled difference images. The angular resolution is controlled in such a way as to approximate the scaling of the angular resolution prescribed for the curvelet system.
3. The directional filterbanks have an inherent subsampling scheme, that makes them orthogonal when employed with perfect reconstruction filters. Combining this with a perfect reconstruction pyramid filterbank, the whole system becomes a perfect reconstruction filterbank with a redundancy factor of 1.333 inherited from the pyramid filter.

The filterbank uses time-domain filtering, leading to linear complexity decomposition and reconstruction algorithms. The effect of combined

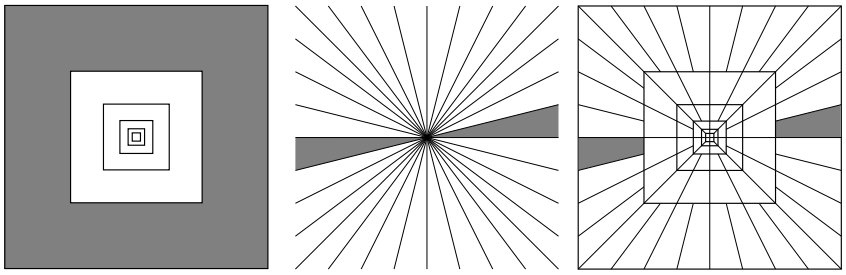

Figure 1.7: Idealised frequency response of the contourlet system. The scaling of the angular resolution is controlled by employing a suitable directional filterbank.

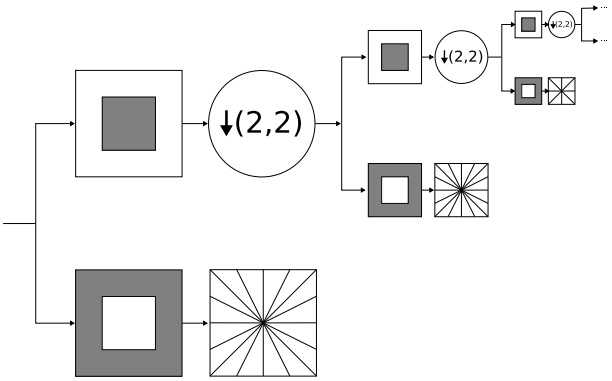

Figure 1.8: Structure of the contourlet decomposition.

bandpass and directional filtering can be inspected in a sample decomposition of a geometric test image in Figure 1.9. The filterbank implementation computes the coefficients of the input image with respect to a family of discrete curvelets or *contourlets*. A small sample of this family is depicted in Figure 1.10, showing that the anisotropic scaling properties of the continuous domain curvelet system are approximately preserved.

In connection with angular resolution, we note a similar phenomenon as for the discrete wedgelet case: Recall that for wedgelets the continuous domain theory prescribed an angular resolution that increases as the scale of the dyadic squares decreases, and that this requirement made little sense for digital images. In the curvelet/contourlet case, the anisotropic scaling amounts to increasing the angular resolution for large frequencies, which cannot be carried out indefinitely for the discrete domain. As we will see in Subsection 1.4.1, contourlets outperform wavelets in the low bit coding area, showing an improved ability to pick out salient image structures,

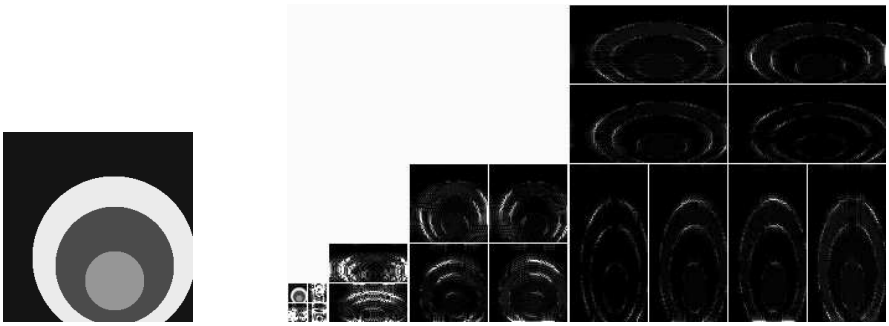

Figure 1.9: Sample image Circles, decomposed by subsequent bandpass and directional filtering.

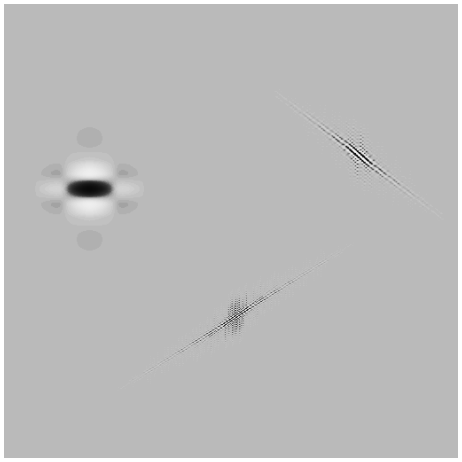

Figure 1.10: A sample of three contourlets of different scales and orientations; the grey-scale is manipulated to improve visibility of the different contourlets. Observe the change in aspect ratios as the scale decreases.

including large scale structures. It seems reasonable to expect that this ability is further improved if we allow the curvelets more orientations in the large scales, which is rather the opposite heuristic to the one proposed for the continuous domain.

1.4 Application to image compression

The novel image representations induced by the methods surveyed in this chapter present potential alternatives to the standard image compression

Figure 1.11: Test images; the usual suspects: Barbara (512×512), Peppers (512×512), Cameraman (256×256), Baboon (512×512) and Circles (256×256) .

schemes, mainly based on wavelet transform. First, in Subsection 1.4.1, we perform some experiments about the practical approximation properties of the representation model. Then in Subsection 1.4.2 we focus on the first attempts in the direction of effective coding-decoding schemes of the underlying information.

1.4.1 Experimental approximation properties

We have conducted tests with real and artificial images to compare the different approximation schemes in terms of coefficients used versus distortion for the reconstructed images. We used standard test images, see Figure 1.11. The wavelet approximations were obtained by the standard matlab implementation, using the db4 filters. The contourlet approximations were obtained using the matlab contourlet toolbox [9], developed by Do and collaborators. For wedgelets we used our implementation available at [34].

The capacity to achieve high theoretical rates of approximation is an important indicator for the potential of a geometrical method in the field of image compression. In the case of two-dimensional locally smooth functions with regular boundaries, we already remarked that the rates obtained with wedgelets or curvelets are of higher order than those induced by the

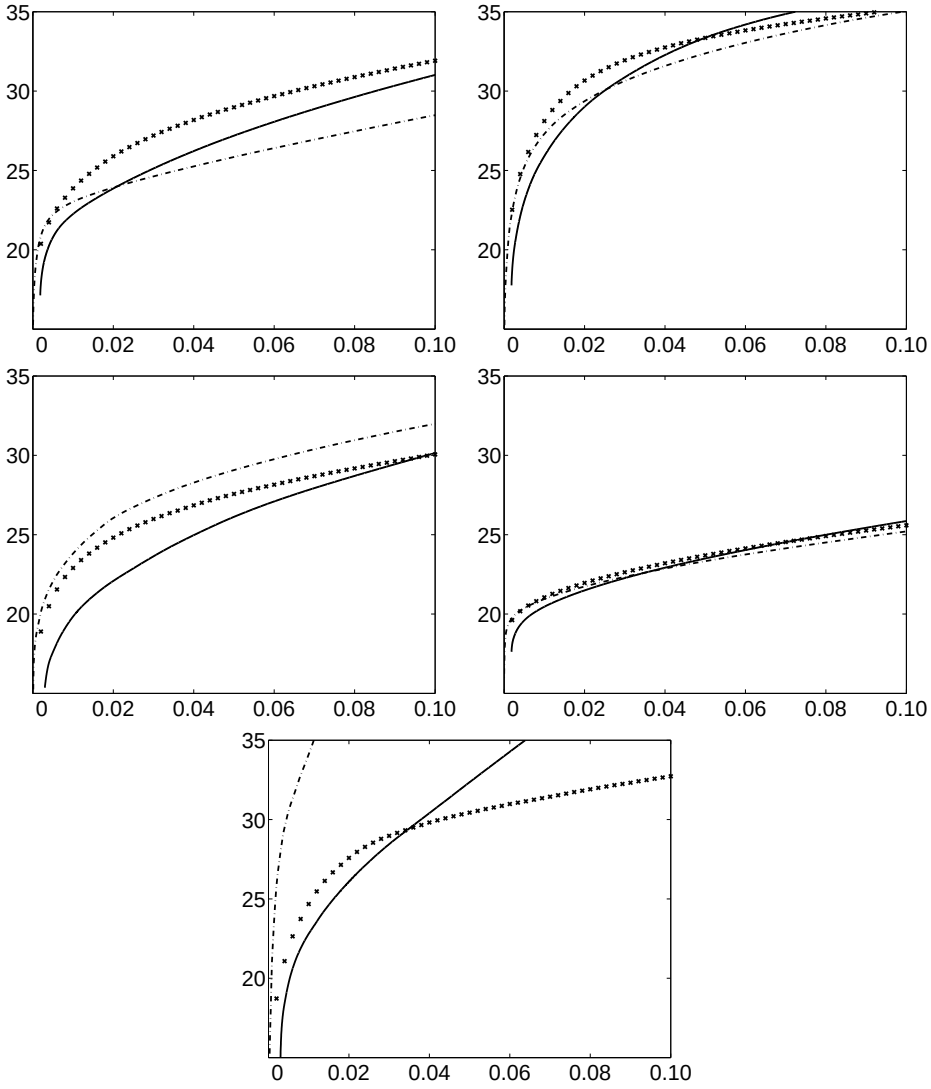

Figure 1.12: Nonlinear approximation behaviour, visualised by plotting coefficients per pixels against PSNR (db). We compare wavelets (solid), contourlets (crosses) and wedgelets (dashed), corresponding to the images, from top left to bottom, Barbara (512×512), Peppers (512×512), Cameraman (256×256), Baboon (512×512) and Circles (256×256).

classical decomposition framework (Fourier decompositions, wavelet frames [25],[26]).

A naive transferal of the approximation results obtained in Section 1.1 would suggest that the new schemes outperform wavelets in the high-bit-rate area, i.e., as the number of coefficients per pixel approaches 1. However, for all images, wavelets have superior approximation rates in these areas. By contrast, contourlets and wedgelets perform consistently better in the low-bit-rate area. Given the fact that the contourlet system has the handicap of a redundancy by a factor of 1.333, and the fact that the approximation is obtained by simple thresholding, the consistently good approximation behaviour of contourlets for extremely small numbers of coefficients is remarkable. Wedgelets, on the other hand, perform best when dealing with images that are of a predominantly geometric nature, such as the cameraman, or the circles. This of course was to be expected. Similarly, the trouble of wedgelets in dealing with textured regions could be predicted beforehand. By contrast, contourlets also manage to represent textured regions to some accuracy, as can be seen in the nonlinear approximation plot for the Barbara image, and in the reconstructions of Baboon and Barbara in Figures 1.13 and 1.14.

Clearly PSNR is not the only indicator of visual quality. Figures 1.13 and 1.14 present reconstructions of our sample images using 0.01 coefficients per pixel. We already remarked that contourlets are superior to wedgelets when it comes to approximating textured regions (cf. Baboon, Barbara). On the other hand, contourlets produce wavelet-like ringing artefacts around sharp edges (cf. Cameraman, Circles). Here wedgelets produce superior results, both visually and in terms of PSNR. As a general rule, the artefacts due to contourlet truncation are visually quite similar to wavelet artefacts. On the other hand, typical wedgelet artefacts come in the form of clearly discernible edges or quantisation effects in the representation of color gradients. To some extent, these effects can be ameliorated by employing a higher order system, such as platelets.

Summarizing the discussion, the results suggest that contourlets and wedgelets show improved approximation behaviour in low-bit-rate areas. Here the improvement is consistent, and somewhat contradictory to the theoretical results which motivated the design of these systems in the first place. Both contourlets and wedgelets are able to well represent the coarse-scale geometric structures inherent in the image. As more and more details are required, wedgelets fail to efficiently approximate textured regions, while in the contourlet case the overall redundancy of the contourlet system increasingly deteriorates the performance.

Figure 1.13: Sample reconstructions using 0.01 coefficients per pixel, for contourlets (left) and wedgelets (right). Top row: Barbara, with contourlets: 23.89 db, wedgelets: 23.02 db; middle row: Peppers, with contourlets: 28.12, wedgelets: 27.84 db; bottom row: Cameraman, with contourlets: 22.90 db, wedgelets: 23.82 db

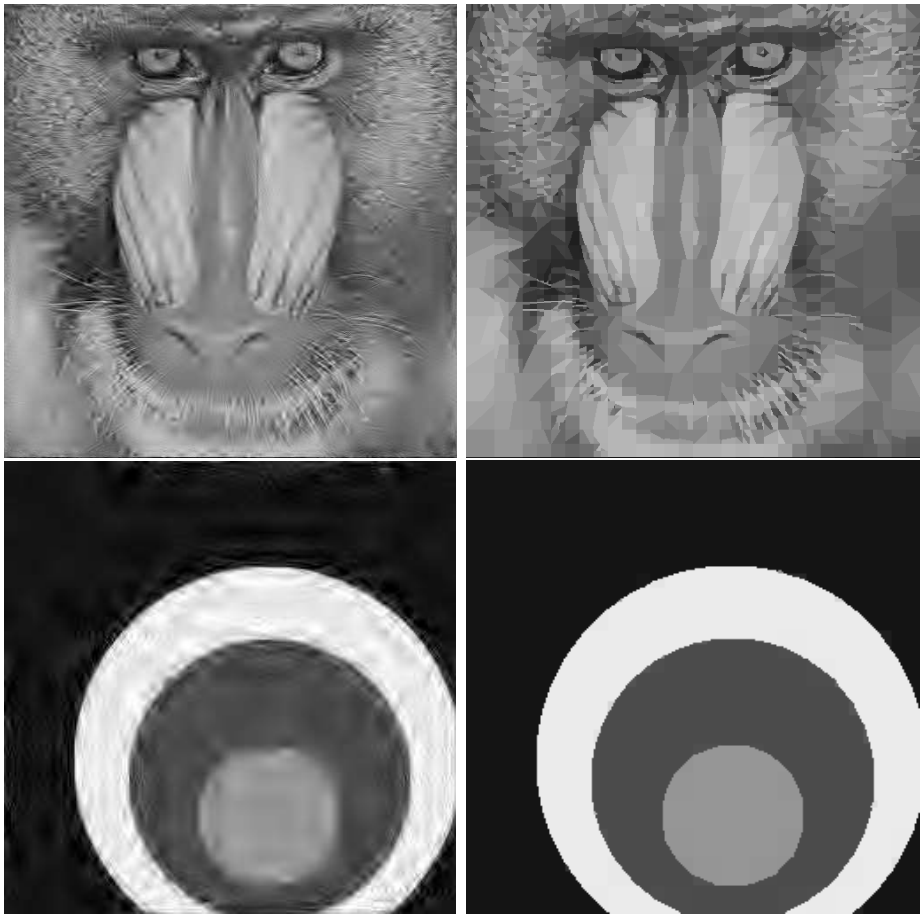

Figure 1.14: Sample reconstructions using 0.01 coefficients per pixel, for contourlets (left) and wedgelets (right). Top row: Baboon, with contourlets: 21.05 db, wedgelets: 20.89 db; bottom row: Circles, with contourlets: 26.56 db, wedgelets: 34.12 db

1.4.2 Coding schemes

As we have already remarked on various occasions, it remains an open question to decide whether these approximation rates constitute an adequate framework for the case of compression of discrete images represented with a discrete set of values. The experiments in Subsection 1.4.1 confirm that due to the discretisation effects, the theoretical approximation rates are not observed in practice, even for reasonably big sizes of images. On the other hand, for representations by very few coefficients, where the discretisation effect is negligible, the asymptotical rates do not bring a very relevant information. It is also obvious that the choice of the L^2-error for measuring the distortion also leads to some undesired artefacts. For instance, this kind of measure also incorporates some noise inherent to natural images [8], and is thus poorly adapted to the human visual systems.

Now let us turn to the other main ingredient required for a complete effective compression scheme, i.e. a suited coding method which captures the remaining structural redundancies in the representation model. In the case of wavelets, tree-based methods [27, 26] allow an efficient coding of coefficients. Apparently, analogs for contourlets have not been developed so far. Recall that a contourlet based coding scheme would also have to make up for the redundancy factor of 1.333 induced by the contourlet decomposition. In any case, we are not aware of an existing implementation of a compression scheme based on contourlets.

Thus we restrict the following discussion to the application of wedgelets to image compression, which is mainly due to the work of Wakin [31, 33]. The first attempts are based on a model mixing cartoon and texture coding [32]. More than for its efficiency, this method is interesting for the sake of understanding the difficulties occurring in the coding of wedge representations of images. The heuristic behind this method consists in considering a natural image as the combination of two independent components, one containing the textures, the other one corresponding to a simple edge cartoon model, containing only the sharp edge information. Then, a separated coding of each component is performed. The cartoon component is treated with the help of wedgelet approximation, whereas the residual error image inferred from this first approximation is coded with wavelets in a classical way. On the following, we focus on the coding of the tree-based wedgelet representation of the cartoon component. The decoder needs to know the structure of the tree (a node of the tree can be either a square leaf, a wedge leaf or subdivided), the wedge parameters for the wedge leaves, and the corresponding quantised optimised constant values for the selected wedges and squares.

Such a *naive coding* of a wedgelet decomposition avoids most ringing

artefacts around the edges, but still suffers of a bad punctual details and texture rendering. In terms of PSNR it remains inferior to wavelet coding (like JPEG2000). Note that this coding scheme is suboptimal, mainly because it does not model the dependencies between neighbouring coefficients, and also because of redundancies between wavelet and wedgelet representations, inherent to this scheme.

For the problem of dependencies between different edges, a possible modelling is the MGM (Multiscale Geometry Model). It relies on a kind of multiscale Markov model for the structural dependencies between wedge orientations and positions; indeed they make use of the probabilities of an angle in a child dyadic square to be selected, conditionally to the optimal angle selected in the parent dyadic square. Note that this model only takes into account the Hausdorff distance between the parents-lines and the children lines. Contrarily to the contextual coding used in JPEG2000 [27], where the Markov models are adaptively updated, this model does not adapt to the image contents, but is rather based on an *ad hoc* assumption concerning the correlation between geometrical structure of parents and children dyadic squares. This joint coding of the wedge parameters allows significant coding gains when compared to an independent coding.

Deriving an efficient compression scheme depends also on the possibility to prune the tree adequately. In (1.13), the penalisation used for the pruning of the tree corresponds to a balance between the distortion in the reconstruction of the image and the complexity of the model measured by the number of pieces retained for the representation of the image. In the compression context, it is interesting to consider a modified penalisation, which takes into account the coding cost. The problem reduces then to the minimisation of a functional of the form

$$H_{\lambda,f}(g,W) = \|f-g\|_2 2 + \lambda[-\log(P(W))] \quad ,$$

where P is an entropy measure depending on some probabilistic assumptions on the data. This measure is used for evaluating the amount of bits required for the coding of the wedge tree and parameters information.

The most interesting method is the W-SFQ (Wedgelet- Space Frequency Quantisation) compression scheme proposed in [33] and based on the use of wedgeprints. The main difference with the previous scheme consists in acting directly in the wavelet coefficients domain, instead of the image domain. The method is mainly an enhancement of the SFQ scheme, which was originally proposed in [36]. SFQ is a zerotree coding where the coefficients clustering are optimised according to a rate-distortion criterion. It is a very efficient method outperforming the standard SPIHT (Set Partitioning Into Hierarchical Trees [26]) in many cases, especially for low bitrates.

It still suffers, however, from the limits of zerotree wavelet coders for the coding of significant wavelet coefficients along edges.

Whereas SFQ considers two possibility for a coded node, either being a zerotree (all its descendants being considered unsignificant) or a significant, coded coefficient, W-SFQ introduces a third possibility, a node can be a wedge, the wavelet coefficients descendants of this node being evaluated from the optimal wedge function. In other words, W-SFQ is a zerotree where the clustering of coefficients is more suited to geometry, with the help of wedge functions. This clustering together with the associated function is also called wedgeprint. Despite the high coding cost of the wedges, the coherence is ensured by the rate-distortion optimisation: a wedge is only chosen when its cost remains low towards the gain in distortion. In [33], an enhanced version of this coding scheme is proposed with some additional ingredients,

- the use of the MGM model which is an efficient tool to code efficiently deeper wedge tilings; a more accurate description of the geometry of a contour than with a single wedgeprint is possible;
- a smoothing parameter for the edges which takes into account blurring artefacts in original image, due to pixelisation;
- a specific coding for textures.

This enhanced version model allows to code the information for less larger wedgeprints. The finer wedgeprints are then coded at low cost thanks to the MGM. Encouraging results were obtained for some natural images. For instance, W-SFQ outperforms SFQ at very low bitrate, for some natural images [33], with poor texture contents, dominated by geometrical structures (and hence closer to the cartoon model).

Note that the MGM model in this context remains relatively rudimentary. Unlike many classical coding schemes, it is a non adaptive model. Furthermore it only takes into account correlations between neighboring scales. One could expect improved compression rates with the help of a modelling of the spatial correlations between wedge parameters of neighbouring wedgeprints.

1.5 Tentative conclusions and suggestions for further reading

The construction and efficient implementation of new multiscale methods for the representation of relevant image structures is an ongoing endeavor. It seems that while the existing constructions already manage to outperform wavelets, at least in certain respects, both theoretically and in practice, a consistent improvement over wavelets, in particular in the domain of

image compression, has not yet been achieved. In this respect one should acknowledge the fact that wavelet compression, as it is incorporated into JPEG2000, is the product of at least ten years of activity that took place between the recognition of the potential of wavelets for image coding, and their eventual inclusion into an image compression standard. As both the diversity of the approaches and their computational complexity –at least in comparison to wavelets– seem to indicate, a further improvement will require considerable effort, and it is hard to predict which method will prove to be the most effective. In this chapter, we have tried to describe some of the approaches that have been developed so far, and for further reading, we refer to our main sources, which were [16, 6, 13, 14, 11]. A closer description of curvelets and ridgelets can also be found in the book [2]. Promising alternatives that we mentioned in Subsection 1.1.3 but did not treat in more detail are bandelets [20, 21], and directionlets [30]. An approach that allows the combination of different approaches, as e.g. wavelets and wedgelets, are footprints (see the discussion of wedgeprints in the previous subsection). A introduction to footprints can be found in [18].

Acknowledgements

HF and FF acknowledge additional funding through the European Research Training Network HASSIP, under the contract HPRN-CT-2002-00285.

References

[1] J.P. Antoine and R. Murenzi. *Two-dimensional directional wavelets and the scale-angle representation,* IEEE Signal Processing **52**, 1996, pp. 259–281.

[2] J.P. Antoine, R. Murenzi, P. Vandergheynst and S.T. Ali. *Two-dimensional Wavelets and Their Relatives.* Cambridge University Press, 2004.

[3] R. Bamberger and M. Smith. *A filter bank for the directional decomposition of images: Theory and design.* IEEE Trans. Signal Proc. 40, 1992, pp. 882-893.

[4] E. Candes and D. Donoho. *Ridgelets: a key to higher-dimensional intermittency ?,* R. Soc. Lond. Philos. Trans. Ser.A Math. Phys. Eng. Sci. **357**,no. 1760, 1999, pp. 2495–2509.

[5] E. Candes and D. Donoho. *Curvelets - A Surprisingly Effective Nonadaptive Representation for Objects with Edges, Curves and Surfaces,*

L. Schumaker et al. (eds), Vanderbilt University Press, Nashville, TN, 1999.

[6] E. Candes and D. Donoho. *New tight frames of curvelets and optimal representations of objects with C^2 singularities,* Commun. Pure Appl. Math. 57, 2004, pp. 219–266.

[7] V. Chandrasekaran, M. Wakin, D. Baron and R. Baraniuk. *Surflets: A Sparse Representation for Multidimensional Functions Containing Smooth Discontinuities* IEEE Symposium on Information Theory, Chicago, IL, (Accepted, June 2004).

[8] A. Cohen, W. Dahmen, I. Daubechies and R. DeVore. *Tree approximation and optimal encoding* J. Appl. Comp. Harm. Anal., **11**, 2001, pp. 192–226.

[9] Website http://www.ifp.uiuc.edu/∼minhdo/software/ , containing a downloadable contourlet toolbox.

[10] L. Demaret, N. Dyn and A. Iske. *Image Compression by Linear Splines over Adaptive Triangulations,* Preprint, University of Leicester, 7 January 2005.

[11] L. Demaret, F. Friedrich, H. Führ and K. Wicker. *Discrete Green's theorem for polygonal domains, with an application to rapid wedgelet approximation.* Preprint 2005, available at http://ibb.gsf.de/preprints.php.

[12] M. Do. *Directional multiresolution image representation,* Thesis, Swiss Federal Institute of Technology, 2001.

[13] M. Do and M. Vetterli. *The contourlet transform: an efficient directional multiresolution image representation,* IEEE Transactions Image on Processing, to appear.

[14] M. Do and M. Vetterli. *Contourlets,* in *Beyond Wavelets*, G. V. Welland ed., Academic Press, 2003.

[15] D. Donoho, M. Vetterli, R. DeVore and I. Daubechies. *Data compression and harmonic analysis.* IEEE Trans. Inf. Th. 44 (6), 1998, pp. 2435–2476.

[16] D. Donoho. *Wedgelets: nearly minimax estimation of edges,* Ann. Statist. 27, no. 3, 1999, pp. 859–897.

[17] D. Donoho and X. Huo. *Beamlets and multiscale image analysis*, Barth, Timothy J. et al.(ed.), *Multiscale and multiresolution methods. Theory and applications.* Springer Lect. Notes Comput. Sci. Eng. 20, 2002, pp. 149–196.

[18] P.L. Dragotti and M. Vetterli. *Wavelet footprints: Theory, Algorithms and Applications,* IEEE Trans. Sig. Proc. 51 (5), 2003, 1307-1323.

[19] F. Friedrich. *Complexity Penalized Segmentations in 2D.* PhD Thesis, Technische Universität München, to appear 2005.

[20] E. LePennec and S. Mallat. *Bandelet representations for image compression,* IEEE Proceedings *International Conference on Image Processing* vol. 1, 2001, 12-14.

[21] E. LePennec and S. Mallat. *Sparse Geometric Image Representations with Bandelets,* to appear in IEEE Transactions on Image Processing, 2004.

[22] Website of LetItWave, http://www.letitwave.fr/

[23] S. Mallat. *A wavelet tour of signal processing.* Academic Press, 1998.

[24] F. Meyer and R. Coifman. *Brushlets: A tool for directional image analysis and image compression,* Appl. Comp. Harm. Analysis 4, 1997, pp. 147–187.

[25] M. Shapiro, *An embedded hierarchical image coder using zerotrees of wavelet coefficients*, IEEE Trans. on Signal Processing **41**, pp. 3445–3462, 1993.

[26] A. Said and W. A. Pearlman. *A new, fast, and efficient image codec based on set partitioning in hierarchical trees*, IEEE Trans. Circuits and Systems for Video Technology, **6**, pp. 243–250, 1996.

[27] D. Taubman, *High performance scalable image compression with EBCOT*, IEEE Trans. on Image Processing, pp. 1158–1170, 2000.

[28] J. Starck, E. Candes and D. Donoho. *The Curvelet Transform for Image Denoising,* IEEE Transactions on Image Processing, 11, 2002, pp. 670–684.

[29] M. Vetterli. *Wavelets, approximation and compression,* IEEE Signal Proc. Mag. **18** (5), 2001, pp. 59–73.

[30] V. Velisavljević, B. Beferull-Lozano, M. Vetterli and P.L. Dragotti. *Directionlets: Anisotropic multi-directional representation with separable filtering.* IEEE Trans. Im. Proc., to appear.

[31] M.B. Wakin. *Image compression using multiscale geometric edge models,* MSc. thesis, Rice University, Houston, 2002.

[32] M.B. Wakin, J.K. Romberg, H. Choi and R.G. Baraniuk. *Image compression using an Efficient Edge Cartoon + Texture Model*, in Proceedings, IEEE Data Compression Conference - DCC'02, Snowbird, April 2002, pp. 43–52.

[33] M.B. Wakin, J.K. Romberg, H. Choi and R.G. Baraniuk. *Wavelet-domain Approximation and Compression of Piecewise Smooth Images,* submitted to IEEE Transactions on Image Processing, November 2004.

[34] Website www.wedgelets.de.

[35] R.M. Willett and R.D. Nowak. *Platelets: A multiscale approach for recovering edges and surfaces in photon-limited medical imaging,* IEEE Transactions on Medical Imaging **22** 2003, pp. 332–350.

[36] Z. Xiong, K. Ramchandran, and M.T. Orchard. *Space-frequency quantization for wavelet image coding,* IEEE Transactions on Image Processing **6**, 1997, pp. 677–693.