BLAC-wavelets:
a multiresolution analysis with non-nested spaces

Georges-Pierre Bonneau[1] Stefanie Hahmann[2] Gregory M. Nielson[3]

Abstract. *In the last five years, there has been numerous applications of wavelets and multiresolution analysis in many fields of computer graphics as different as geometric modelling, volume visualization or illumination modelling. Classical multiresolution analysis is based on the knowledge of a nested set of functional spaces in which the successive approximations of a given function converge to that function, and can be efficiently computed. This paper first proposes a theoretical framework which enables multiresolution analysis even if the functional spaces are not nested, as long as they still have the property that the successive approximations converge to the given function. Based on this concept we finally introduce a new multiresolution analysis with exact reconstruction for large data sets defined on uniform grids. We construct a one-parameter family of multiresolution analyses which is a blending of Haar and linear multiresolution.*

1. Introduction

With a multiresolution analysis one can represent a given function at multiple levels of detail. The loss of detail in each level is stored into the so-called wavelet coefficients (detail coefficients). Wavelets are basis functions encoding the difference between two successive levels of representation. Multiresolution analysis consists in other words in a decomposition of a given function over a coarse subdivision of the domain together with a sequence of wavelet coefficients and allows an exact reconstruction.

Multiresolution analysis based on wavelets is a powerful tool for handling with large data sets. It allows an efficient representation by means of compression and fast computations. One can find its origins in numerical analysis and signal decomposition. But in the last five years, they became more and more popular in many fields of computer graphics as different as image processing, geometric modelling, volume visualization or illumination modelling.

Compression of large data sets like images can be done by removing small wavelet coefficients. The number of removed coefficients depends on the error bound the resulting approximation should be within. With a multiresolution analysis it is easy to perform *progressive displaying* of complex scenes. While adding progressively the wavelet coefficients the image is displayed more and more in detail. One does not need to load and display a whole large image all at once. Multiresolution analysis is also suitable for *level-of-detail control* in rendering systems. One can go smoothly from a very coarse approximation of an object far away from the viewer to more detailed representations more the viewer approaches the object. All these application examples and more can be found in [Eck95].

Classical multiresolution analysis is based on the knowledge of a nested set of functional spaces in which the successive approximations of a given function converge to that function, and can be efficiently computed.

This paper first proposes a theoretical framework which enables multiresolution analysis even if the functional spaces are not nested, as long as they still have the property that the successive approximations converge to the given function. The purpose of this framework is to introduce a new multiresolution analysis of large data sets defined on uniform grids. For these sets two well known wavelet bases, the Haar and the linear basis, can be used. Haar wavelets allow local analysis formulas but they are not continuous. Linear wavelets are continuous but their regularity can be a drawback if the analyzed data have many discontinuities. The theoretical framework introduced in this paper enables to find a one-parameter family of wavelet bases which goes smoothly from the Haar to the linear basis. The user can choose his own compromise between regularity and locality properties.

[1] Laboratoire LIMSI-CNRS, BP 133, F-91403 Orsay, France

[2] Laboratoire LMC-IMAG, Université INPG Grenoble, BP 53, F-38041 Grenoble, France

[3] Computer Science Department, Arizona State University, Tempe, AZ 85287-5406, USA

2. Theoretical Background

The purpose of this section is to recall briefly the basics of a classical multiresolution analysis, omitting technical details, to introduce our new multiresolution framework, and to point out links between both theories. Multiresolution analysis based on wavelets, as first introduced by Mallat in [Mall89], leads to a hierarchical scheme for the computation of the wavelet coefficients. The use of fully orthogonal wavelet bases generally implies globally supported wavelets. But the infinite support of such wavelets is usually a disadvantage for practical implementations. This is the reason why we focus on semi-orthogonal wavelet bases.

2.1 Classical multiresolution analysis

Let Ω be some domain and $L_2(\Omega)$ the set of all functions with finite energy over Ω. Classical multiresolution analysis is based on the knowledge of a *nested* set of subspaces V_n of $L_2(\Omega)$.

$$V_n \subset V_{n+1} \tag{1}$$

It is also required that

$$\bigcup_n V_n \text{ is dense in } L_2(\Omega)\ , \tag{2}$$

so that for any given function f in $L_2(\Omega)$, the successive approximations f_n of f in V_n will converge. Property (1) ensures that f_{n+1} is always a better approximation of f than f_n. The *wavelet spaces* W_n are introduced to capture the loss of detail between f_{n+1} and f_n. In other words there must exist some function g_n in W_n so that $f_{n+1} = f_n + g_n$. This is the case if we choose W_n as the orthogonal complement of V_n in V_{n+1}.

$$V_{n+1} = V_n \oplus W_n \tag{3}$$

In order to compute the approximations f_n and the details g_n, one has to introduce basis functions for the spaces V_n and W_n. In this section, we won't state more precise any index set. This will be done in section 3. The basis functions (φ_i^n) of V_n are called *scaling functions*. The basis functions (ψ_j^n) of W_n are called *wavelet functions*.

From (1) and (3) follows that there exist some matrices $G^n = (g_{ik}^n)$ and $H^n = (h_{ik}^n)$, so that

$$\varphi_i^n = \sum_k g_{ik}^n\ \varphi_k^{n+1} \tag{4}$$

$$\psi_j^n = \sum_k h_{jk}^n\ \varphi_k^{n+1} \tag{5}$$

(4) is known as the refinement equation. Another consequence of (3) is that the functions φ_i^n and ψ_j^n are mutually orthogonal, and therefore we have

$$\sum_k \sum_l g_{ik}^n\ h_{jl}^n < \varphi_k^{n+1}, \varphi_l^{n+1} >= 0 \tag{6}$$

The orthogonality equation (6) is used to compute the matrix H^n and hence the wavelet functions (ψ_j^n) from a given set of scaling functions (φ_i^n) and (φ_k^{n+1}).
From (3) follows also that the finer scaling functions (φ_k^{n+1}) are linear combinations of the coarser scaling functions (φ_i^n) and the wavelet functions (ψ_j^n):

$$\varphi_k^{n+1} = \sum_k \overline{g_{ki}^n}\ \varphi_i^n + \sum_k \overline{h_{kj}^n}\ \psi_k^n$$

Given an approximation $f_{n+1} = \sum_k x_k^{n+1} \varphi_k^{n+1}$ of f in V_{n+1}, the coarser approximation $f_n = \sum_i x_i^n\ \varphi_i^n$ and the details $g_n = \sum_j y_j^n\ \psi_j^n$ are computed by

$$x_i^n = \sum_k \overline{g_{ki}^n}\ x_k^{n+1} \tag{7}$$

$$y_j^n = \sum_k \overline{h_{kj}^n}\; x_k^{n+1} \tag{8}$$

(7) and (8) are called *analysis formulas*.
Computing the finer smoothing coefficients (x_k^{n+1}) from the coarser one (x_i^n) and the detail coefficients (y_j^n) is done in the following way:

$$x_k^{n+1} = \sum_i g_{ik}^n\; x_i^n + \sum_j h_{jk}^n\; y_j^n \tag{9}$$

This equation is known as *synthesis formula.*

The complete analysis process, called *filter bank*, is represented by

$$\begin{array}{ccccccccc}
(\mathbf{x}_k^{n+1}) & \longrightarrow & (\mathbf{x}_k^{n}) & \longrightarrow & & \cdots & (\mathbf{x}_k^{1}) & \longrightarrow & (\mathbf{x}_k^{0}) \\
& \searrow & & \searrow & & & & \searrow & \\
& & (\mathbf{y}_j^{n}) & & (\mathbf{y}_j^{n-1}) & \cdots & & & (\mathbf{y}_j^{0})
\end{array}$$

The complete synthesis process consists of recursively computing the finest smoothing coefficients (x_k^{n+1}) from the coarsest (x_i^0) and the detail coefficients at each scale, by $n+1$ applications of (9).

2.2 Multiresolution analysis with non-nested spaces

The choice of the scaling functions is the heart of the multiresolution analysis introduced in 2.1. This choice determines the approximation spaces V_j. If the convergence condition (2) is quite intuitive since we want the approximation f_n to converge, it should be possible to define a multiresolution framework even if the condition of nested spaces (1) is not verified. In fact this condition, or equivalently the refinement condition (4) is very restrictive. It forces the scaling functions at all levels to have the same regularity. This is incompatible with the BLAC-scaling functions defined in section 3 of this paper.

The idea applied in section 3 to do multiresolution analysis without the refinement condition, is basically to associate to each scaling function at the level n, a linear combination of scaling functions of the finer level $n+1$, which is not equal (since it's not possible), but close to it. In other words, we replace the refinement condition by an *approximated refinement condition:*

$$\varphi_i^n \simeq \widetilde{\varphi}_i^n = \sum_k g_{ik}^n \varphi_k^{n+1} \tag{10}$$

The choice of the approximation depends on the application, and therefore it will be stated more precisely in the next section. The new approximation space $\widetilde{V}_n$ spanned by the functions $(\widetilde{\varphi}_i^n)$ is close to V_n, and it is also included in V_{n+1}:

$$V_n \simeq \widetilde{V}_n \subset V_{n+1}\ .$$

The wavelet space W_n now captures the loss of detail between $\widetilde{V}_n$ and V_{n+1}:

$$V_{n+1} = \widetilde{V}_n \oplus W_n\ . \tag{11}$$

The orthogonality equation (6) used to compute the wavelets is still the same, but it means now that ψ_i^n and $\widetilde{\varphi}_i^n$ (instead of φ_i^n) are mutually orthogonal.

The analysis and synthesis formulas (7), (8) and (9) are the same, but it corresponds now to an approximation in two steps:

$$\textbf{Analysis:}\quad \mathbf{f_{n+1}} \overset{\text{approx.}}{\longrightarrow} \widetilde{f}_n = \textstyle\sum_i x_i^n\, \widetilde{\varphi}_i^n \overset{\text{approx.}}{\longrightarrow} \mathbf{f_n} = \textstyle\sum_i x_i^n\, \varphi_i^n$$
$$\searrow \qquad \mathbf{g_n}$$

Figure 1 gives a geometric interpretation of one step of analysis and synthesis in the classical and new multiresolution framework.

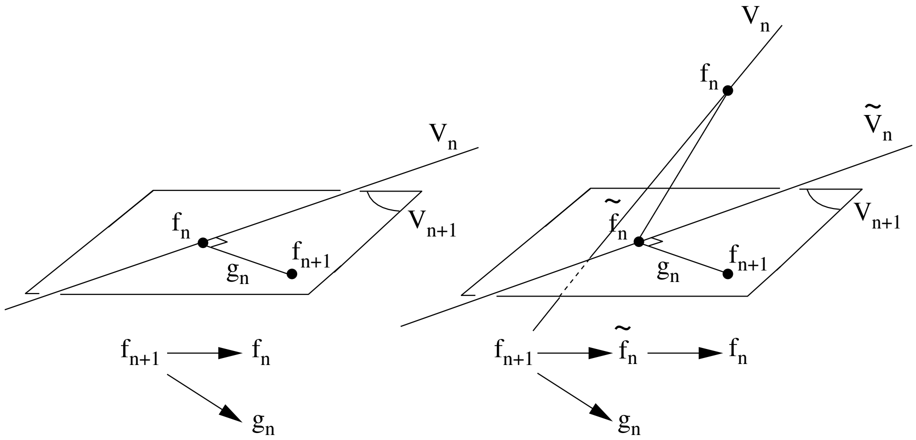

Figure 1: *The classical and new multiresolution framework*

This new multiresolution framework is applied in section 3, where we will need to handle scaling functions with different regularities.

3. BLAC multiresolution

This section deals with the multiresolution analysis of data defined on uniform grids. Since we want our algorithm to be applied to very large data sets we keep on low degree scaling and wavelet functions. Our starting point while developing BLAC wavelets was to look at the Haar and linear biorthogonal wavelets [Sto95a,b], [Fin94]. This last kind of wavelets have some regularity, they are continuous, which is not the case of the piecewise constant Haar wavelets. But the regularity of the basis functions is a good point only if the data to be analyzed is also continuous. To illustrate this fact, figure 2 shows the result of one step of analysis on a 16 point data set, all but one equal to zero, by Haar and linear wavelets. We can see that the analysis by Haar wavelets yields only two non zero coefficients, while the analysis by linear wavelets gives much more non zero coefficients. Since compression of the data follows from omitting all coefficients which are near to zero, the consequence of this is that linear wavelets yield bad compression ratio in the area where the data is discontinuous.

The purpose of the BLAC wavelets is to find a compromise between the locality of the analysis (which is perfect for the Haar wavelets) and the regularity of the approximation (which is much better for the linear wavelets). Therefore BLAC stands for *Blending of Linear And Constant*. The graph on the right of figure 2 shows the result of one analysis step by a family of BLAC wavelets near from the Haar wavelets.

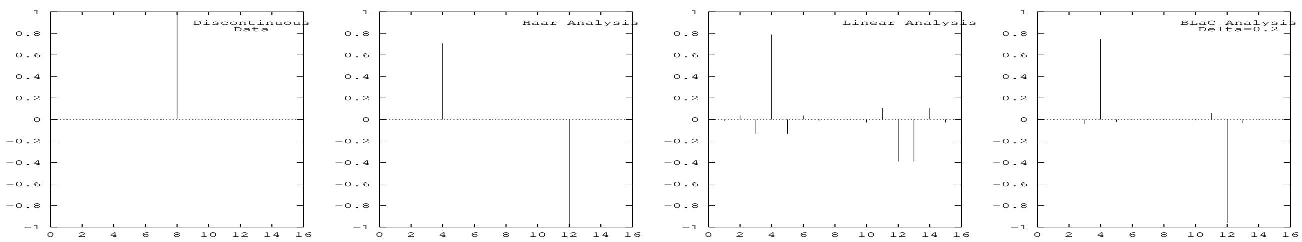

Figure 2: *The test data set and comparison of Haar, linear and BLAC analysis*

In section 3.1 we describe the BLAC scaling and wavelet functions. Section 3.2 gives some implementation remarks and compare the results of the Haar, BLAC and linear multiresolution analysis on some examples.

3.1 The BLAC scaling and wavelet functions

BLAC scaling functions. The first step in defining a multiresolution analysis as introduced in section 2.2, is the choice of the scaling functions. Since we wanted to find a blending between Haar and linear wavelets, the following fundamental scaling function φ was chosen:

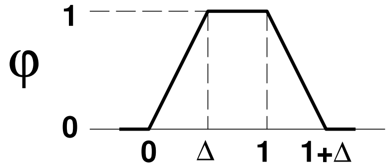

Figure 3

The scaling function φ_i^n is built from φ by dilation by a factor 2^n, and translation of $(i\ 2^{-n})$:

$$\varphi_i^n(x) = \varphi\big(2^n(x + i\ 2^{-n})\big)\ .$$

The fundamental function φ depends on the blending factor Δ: for $\Delta = 0$, (φ_i^n) are the usual Haar scaling functions, and for $\Delta = 1$ they are the linear scaling functions. Figure 4 shows the function φ for various values of Δ between 0 and 1.

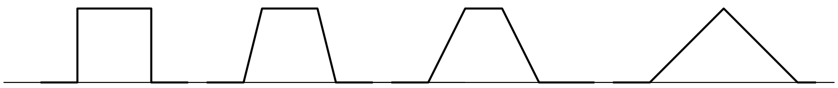

Figure 4: *The BLAC scaling functions*

In order to have end-point interpolating scaling functions, we define the first and last scaling functions as follows:

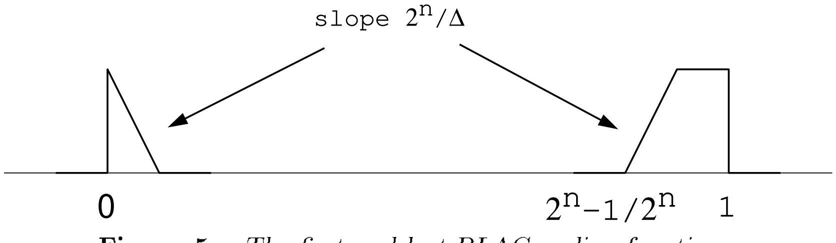

Figure 5: *The first and last BLAC scaling functions*

The approximation space V_n has the dimension 2^n+1, and is spanned by the scaling functions $\varphi_0^n, \cdots, \varphi_{2^n}^n$.

Approximated refinement equations. The scaling functions of the level of resolution $n+1$ have a slope of $2^{n+1}/\Delta$. They are less regular than the scaling functions at one coarser level of resolution, since they "climb" to 1 more quickly.

The function φ_i^n is a so-called L-Lipschitz function with regularity $L = 2^n/\Delta$. As we pointed out in section 2.2, the fact that the scaling functions at different level of resolution have different regularity implies that no refinement equation can be found. Instead of that, an approximation $\widetilde{\varphi}_i^n$ of the coarser scaling function φ_i^n as linear combination of the finer scaling functions has to be chosen (see section 2.2, equation (10)). For Haar scaling functions, the exact refinement equation is $\varphi_i^n = \varphi_{2i-1}^{n+1} + \varphi_{2i}^{n+1}$, and in the linear case it is $\varphi_i^n = \frac{1}{2}\varphi_{2i-1}^{n+1} + \varphi_{2i}^{n+1} + \frac{1}{2}\varphi_{2i+1}^{n+1}$. Since we want to blend between Haar and linear, we choose $\widetilde{\varphi}_i^n$ to be a linear combination of φ_{2i-1}^{n+1}, φ_{2i}^{n+1}, φ_{2i+1}^{n+1}. And since we want the analysis scheme to preserve constant functions we search the following approximation of φ_i^n:

$$\varphi_i^n \simeq \widetilde{\varphi}_i^n = \alpha\varphi_{2i-1}^{n+1} + \varphi_{2i}^{n+1} + (1-\alpha)\varphi_{2i+1}^{n+1} \tag{12}$$

The last restriction on the coefficient of the linear combination ensures that constant functions are included in the approximation space $\widetilde{V}^n$, as defined in 2.2, and thus that the analysis scheme will preserve them. The constant α is calculated to minimize the L_2 distance between φ_i^n and $\widetilde{\varphi}_i^n$:

$$\|\varphi_i^n - \widetilde{\varphi}_i^n\|_{L_2(I\!R)} \longrightarrow \min$$

The minimum is reached if α is the solution of the following linear equation:

$$< \alpha\varphi_{2i-1}^{n+1} + \varphi_{2i}^{n+1} + (1-\alpha)\varphi_{2i+1}^{n+1} - \varphi_i^n\,, \varphi_{2i-1}^{n+1} - \varphi_{2i+1}^{n+1} >= 0 \tag{13}$$

(13) proves that α is independent of the level of resolution n. For $\Delta = 0$, the solution α is equal to 1, and the equation (12) reduces to the Haar refinement equation. For $\Delta = 1$, the solution is $\alpha = 1/2$, and (12) becomes the linear refinement equation. Figure 6 illustrates the approximated refinement equation (12), and compares it to the refinement equation for the Haar and linear scaling functions.

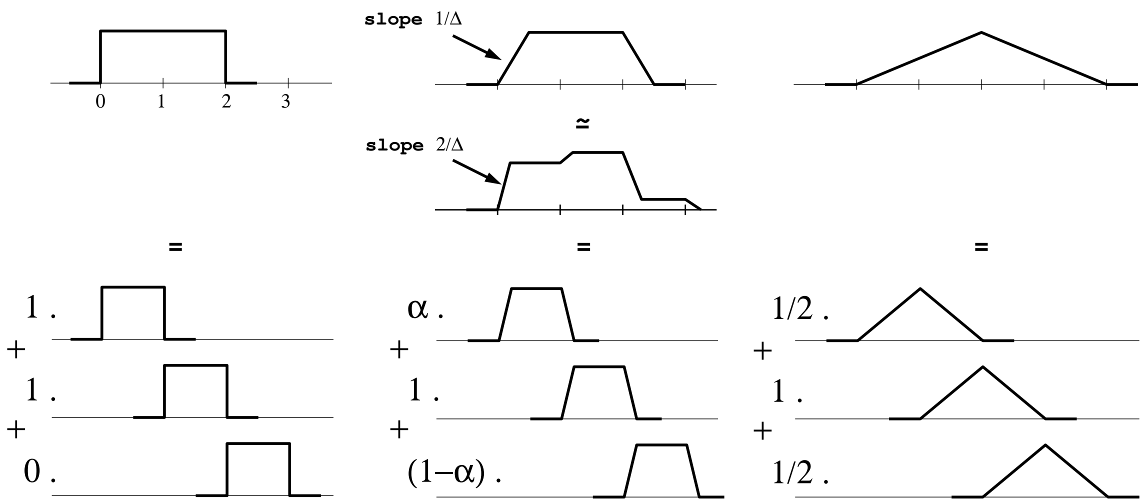

Figure 6: *The scaling functions and the refinement equation: Haar, BLAC, linear*

The approximated refinement equations for the first and last scaling functions involve only two finer scaling functions:

$$\varphi_0^n \simeq \tilde{\varphi}_0^n = \varphi_0^{n+1} + (1-\alpha)\varphi_1^{n+1}$$
$$\varphi_{2^n}^n \simeq \tilde{\varphi}_{2^n}^n = \alpha\varphi_{2^{n+1}-1}^{n+1} + \varphi_{2^{n+1}}^{n+1}$$

BLAC wavelets. Once the scaling functions and the approximated refinement equation is fixed, the wavelet spaces W_n are also fixed by (11). The wavelet functions must be a base of this space. Obviously there is an infinite number of bases, and as in the Haar and linear case we choose the functions which have a minimal support. This goal is achieved if we impose ψ_i^n to be a linear combination of five successive scaling functions:

$$\psi_i^n = a\ \varphi_{2i-1}^{n+1} + b\ \varphi_{2i}^{n+1} + c\ \varphi_{2i+1}^{n+1} + d\ \varphi_{2i+2}^{n+1} + e\ \varphi_{2i+3}^{n+1}$$

The coefficients a, b, c, d, e are computed by the fact that ψ_i^n must be orthogonal to each function of the approximation space $\widetilde{V}_n$. Since there are exactly four basis functions of this space whose support intersects the support of ψ_i^n, the orthogonality condition reduces to:

$$< \psi_i^n, \widetilde{\varphi}_j^n >= 0 \quad \text{with} \quad j = i-1, i, i+1, i+2 \tag{14}$$

(14) is a linear homogeneous system with four equations in the five unknowns a, b, c, d, e. It is uniquely solved if we impose the L_2-norm of the wavelet to be equal to 1, as usual in multiresolution analysis:

$$\int |\psi_i^n|^2 = 1$$

Note that the coefficients a, b, c, d, e are also independent of the resolution level n. Figure 7 shows one BLAC wavelet for various values of Δ.

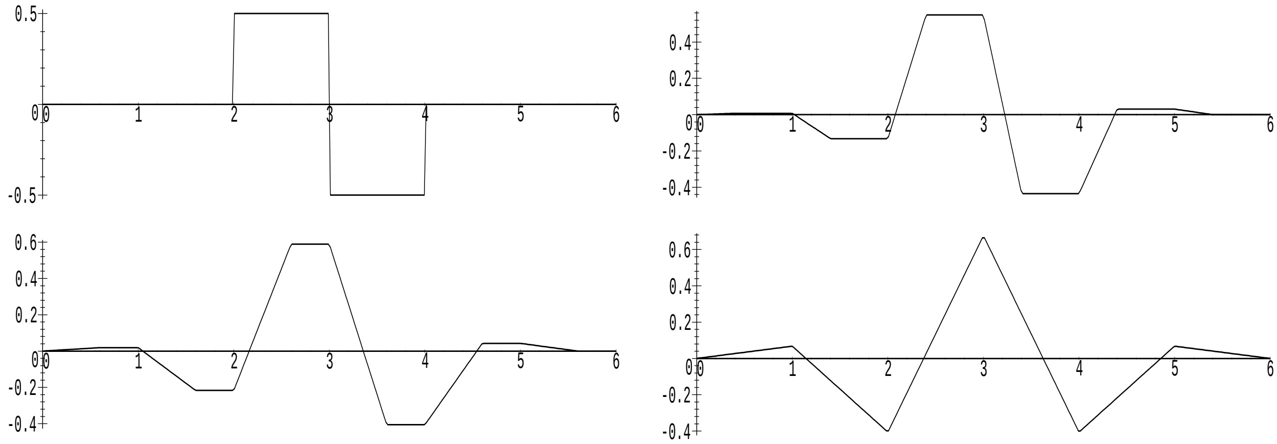

Figure 7: *The BLAC wavelets* †

The first and last wavelet functions can be defined as the linear combination of only four finer scaling functions:

$$\begin{aligned}
\psi_0^n &= b_1\varphi_0^{n+1} + c_1\varphi_1^{n+1} + d_1\varphi_2^{n+1} + e_1\varphi_3^{n+1} \\
\psi_{2^n-1}^n &= a_2\varphi_{2^{n+1}-3}^{n+1} + b_2\varphi_{2^{n+1}-2}^{n+1} + c_2\varphi_{2^{n+1}-1}^{n+1} + d_2\varphi_{2^{n+1}}^{n+1}
\end{aligned}$$

The coefficients $b_1, c_1, d_1, e_1, a_2, b_2, c_2, d_2$ are solution of the following homogeneous systems:

$$\begin{aligned}
&< \psi_0^n, \tilde{\varphi}_j^n >= 0, \qquad j = 0, 1, 2 \\
&< \psi_{2^n-1}^n, \tilde{\varphi}_j^n >= 0, \qquad j = 2^n - 2, 2^n - 1, 2^n
\end{aligned}$$

These coefficients are uniquely defined if we choose the first and last wavelet functions to be normalized:

$$\int |\psi_0^n|^2 = 1$$
$$\int |\psi_{2^n}^n|^2 = 1$$

† *An MPEG movie showing the BLAC wavelet going smoothly from Haar to linear is available via anonymous ftp at ftp.limsi.fr/pub/bonneau/wav.mpg*

3.2 Implementation

The analysis and synthesis scheme (7), (8) and (9) have to be carefully implemented since we want our algorithm to work on large data sets. The results of section 3.1 give the coefficients for the synthesis formula (9). Normally the coefficients for the analysis are computed from those of the synthesis by the inverse of a $(2^{n+1}+1) \times (2^{n+1}+1)$ matrix. But we will see that by using orthogonality properties, the computation of one analysis step can be reduced to the inversion of two smaller symmetric positive definite banded matrices.

Given the approximation f_{n+1} of f at the resolution level $n+1$:

$$f_{n+1} = \sum_{i=0}^{2^{n+1}} x_i^{n+1} \, \varphi_i^{n+1} ,$$

the problem is to find the coarser approximation $\widetilde{f}_n = \sum_{i=0}^{2^n} x_i^n \, \widetilde{\varphi}_i^n$ and the detail $g_n = \sum_{j=0}^{2^n-1} y_j^n \, \psi_j^n$. Since g_n is the error between f_{n+1} and $\widetilde{f}_n$, the following equation needs to be verified

$$\sum_{i=0}^{2^{n+1}} x_i^{n+1} \, \varphi_i^{n+1} = \sum_{i=0}^{2^n} x_i^n \, \widetilde{\varphi}_i^n + \sum_{j=0}^{2^n-1} y_j^n \, \psi_j^n \tag{15}$$

To find the smooth coefficients (x_i^n), we proceed by taking the scalar product of $\widetilde{\varphi}_k^n$ with both sides of (15). Since the wavelets (ψ_j^n) are orthogonal to the scaling function $\widetilde{\varphi}_k^n$, we get

$$< \sum_{i=0}^{2^{n+1}} x_i^{n+1} \, \varphi_i^{n+1}, \widetilde{\varphi}_k^n >= \sum_{i=0}^{2^n} x_i^n < \widetilde{\varphi}_i^n, \widetilde{\varphi}_k^n > \tag{16}$$

Thus we see that the smooth coefficients (x_i^n) are the solution of a linear system whose matrix has the elements $< \widetilde{\varphi}_i^n, \widetilde{\varphi}_k^n >$ and is known as Gram-Schmidt matrix. This matrix has very good properties: it is symmetric, positive definite, and it is banded. We can therefore use the Cholesky algorithm to inverse the system (16). This algorithm is numerically stable for high dimension matrices. This is crucial for the analysis of large data sets. Moreover, the Cholesky algorithm preserves the band structure of the matrix, so that the cost of the inversion of (16) is only of $O(d)$, where d is the size of the data set.

Analogously, the detail coefficients (y_j^n) are the solution of the following linear system:

$$< \sum_{i=0}^{2^n} x_i^{n+1} \, \varphi_i^{n+1}, \psi_k^n >= \sum_{j=0}^{2^n-1} y_j^n < \psi_j^n, \psi_k^n > \quad .$$

This system can also be solved by a Cholesky algorithm for band matrices. Thus the cost of the computation of one analysis step is simply a $O(d)$, where d is the size of the data set.
The scalar products of the scaling functions, from which the elements of the matrices can be computed, are given in the appendix.

3.3 Examples

The examples of figure 8 shows at the top the result of the compression of the 256×256 *lena* image by a factor of 96% with a BLAC multiresolution analysis for three different values of Δ. On the top left we used $\Delta = 0$ (e.g. Haar analysis), in the middle $\Delta = 0.5$ was used and to the top right $\Delta = 1$ (e.g. linear analysis) was employed. As a consequence of the regularity of the linear wavelets, fuzzy areas appear where the image has sudden discontinuities (see f.ex. along the upper border of the hat). This is what we could expect from the introduction of section 3: too much regularity implies bad compression ratio where the data is discontinuous. These fuzzy effects are less visible for $\Delta = 0.5$ and they totally disappear with $\Delta = 0$. On the other hand, the three images at the bottom, which are the results of an edge detection algorithm applied to the top images, clearly show that the approximation becomes smoother while Δ increases. For $\Delta = 0$, the contours even follow the support of the wavelet functions with the biggest coefficients.

Figure 8: *Lena* ‡

To conclude this article, we see that BLAC multiresolution analysis enables the user to choose his own compromise between regularity of the basis functions and locality of the analysis, or what is equivalent, between the *smoothness* of the approximation, and its *sharpness*.

References

1. **G.P. Bonneau; S. Hahmann; G.M. Nielson.** BLAC-wavelets: a multiresolution analysis with non-nested spaces, *submitted to EUROGRAPHIC'96, january 1996*
2. **A. Cohen; I. Daubechies; J. Feauveau.** Bi-orthogonal bases of compactly supported wavelets, *Comm. Pure Appl. Math. 45, 485–560 (1992)*
3. **A. Cohen.** *Ondelettes et Traitement Numérique du Signal*, Eds Masson, Paris (1992).
4. **M. Eck; T. DeRose, et al.** Multiresolution Analysis of Arbitrary Meshes, *Proceedings of SIGGRAPH'95. In Computer Graphics, Annual conference Series, (1995)*
5. **A. Finkelstein; D. Salesin** Multiresolution Curves, *Proc. of SIGGRAPH'94, In Computer Graphics, Annual conference Series, (1994)*
6. **S. Mallat.** A Theory for Multiresolution Signal Decomposition: The Wavelet Representation, *IEEE Trans. Pattern Anal. and Machine Intel. 11, 674–693, (1989)*
7. **E. Stollnitz; T. DeRose; D. Salesin.** Wavelets for Computer Graphics: A Primer, Part 1, *IEEE Computer Graphics & Applications 15 (3??), 76–84, (1995)*
8. **E. Stollnitz; T. DeRose; D. Salesin.** Wavelets for Computer Graphics: A Primer, Part 2, *IEEE Computer Graphics & Applications 15 (4??), 75–85, (1995)*

‡ Available via anonymous ftp at ftp.limsi.fr/pub/bonneau/lena_fig.rgb in sgi-format.