



HAL
open science

Mean Square Error Approximation for Wavelet-based Semiregular Mesh Compression

Frédéric Payan, Marc Antonini

► **To cite this version:**

Frédéric Payan, Marc Antonini. Mean Square Error Approximation for Wavelet-based Semiregular Mesh Compression. *IEEE Transactions on Visualization and Computer Graphics*, 2006, 12 (4), pp.649-657. hal-00264508

HAL Id: hal-00264508

<https://hal.science/hal-00264508>

Submitted on 17 Mar 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Mean Square Error Approximation for Wavelet-Based Semiregular Mesh Compression

Frédéric Payan and Marc Antonini

Abstract—The objective of this paper is to propose an efficient model-based bit allocation process optimizing the performances of a wavelet coder for semiregular meshes. More precisely, this process should compute the best quantizers for the wavelet coefficient subbands that minimize the reconstructed mean square error for one specific target bitrate. In order to design a fast and low complex allocation process, we propose an approximation of the reconstructed mean square error relative to the coding of semiregular mesh geometry. This error is expressed directly from the quantization errors of each coefficient subband. For that purpose, we have to take into account the influence of the wavelet filters on the quantized coefficients. Furthermore, we propose a specific approximation for wavelet transforms based on lifting schemes. Experimentally, we show that, in comparison with a “naïve” approximation (depending on the subband levels), using the proposed approximation as distortion criterion during the model-based allocation process improves the performances of a wavelet-based coder for any model, any bitrate, and any lifting scheme.

Index Terms—Weighted mean square error (MSE), biorthogonal wavelet, lifting scheme, butterfly scheme, bit allocation, geometry coding, semiregular meshes.

1 INTRODUCTION

WAVELETS are now frequently exploited to perform efficient compression. Based on multiresolution analysis, not only do wavelet coders achieve better compression rates [1], [2], [3], [4] than methods based on signal quantization, but they also make the progressive transmission, the adaptive displaying, or the level of details control easier.

Compression algorithms always attempt to optimize the trade-off between rate and distortion. Because of the multiresolution representation of the transformed data, a low frequency signal, and several levels of details (see Fig. 1), the rate-distortion optimization problem for a wavelet coder amounts to dispatch pertinently the bits across the different subbands, in order to obtain the highest quality of the reconstructed signal for a specific bitrate. One relevant method to solve this problem is to include a bit allocation process in the compression algorithm. This process should compute the best quantizers for all the coefficient subbands that minimize the reconstructed mean square error (MSE) due to data quantization for one specific target bitrate.

1.1 Goal and Contributions

The main objective of our current work is to design such an allocation process for a wavelet-based geometry coder of semiregular meshes [5], [6]. In particular, we aim to design a fast and low complex algorithm. For this purpose, some previous works have proposed to express the MSE relative to a quantized signal directly from the MSE of each

subband of wavelet coefficients by taking into account the influence of the synthesis wavelet filters on the quantized coefficients [7], [8], [9], [10], [11]. Unfortunately, these approximations have been developed and exploited only on canonical sampling grids (in image or video processing for instance). Moreover, the proposed formulations are always adapted for “classical” wavelet transforms, i.e., defined by low pass and high pass filters.

Therefore, the contribution of this paper is the development of an approximation of the MSE relative to the reconstructed mesh geometry (that is, a signal sampled on a triangular edge lattice) across a wavelet coder. We show precisely that this specific MSE can be expressed as a weighted sum of the MSE of each coefficient subband. We also observe that, like in the previous works on the canonical sampling grids, the weighting on a triangular edge lattice depends on the wavelet transform used. Furthermore, as our work is only focused on wavelet transforms based on lifting schemes, we also show that the weighting can be expressed only in a function of the prediction and update operators of lifting schemes.

We would like to draw the readers’ attention to the fact that we already used this MSE in a previous paper [6]. However, in [6], its computation was not developed and only focused the unlifted butterfly transform [4]. So, this paper has three specific objectives. First, this manuscript complements other works by the authors. In particular, it fills in details about the computation of the weighting and provides the relative numerical values used in [6]. Moreover, this paper shows that this weighting can be applied easily to any lifting scheme and used for any kind of (semi)regular meshes. Finally, we experimentally show the interest of using this weighting in a wavelet-based geometry coder by evaluating the coding gains relative to its use during the bit allocation.

The remainder of this paper is organized as follows: Section 2 presents the background and the problem

• The authors are with the Laboratoire I3S—UMR 6070 CNRS—Université de Nice-Sophia Antipolis 2000, route des Lucioles—06903 Sophia Antipolis, France. E-mail: {fpayan, am}@i3s.unice.fr.

Manuscript received 5 Dec. 2005; accepted 30 Jan. 2006; published online 10 May 2006.

For information on obtaining reprints of this article, please send e-mail to: tcvg@computer.org, and reference IEEECS Log Number TVCG-0217-1205.

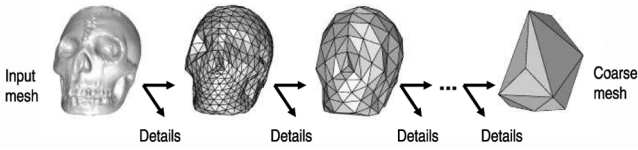


Fig. 1. Multilevel decomposition of a semiregular mesh.

statement of our work. Section 3 develops the MSE approximation of a triangular mesh geometry. Then, Section 4 provides the numerical values of the weighting for the Butterfly-based lifting schemes. Finally, we show experimentally the interest of using the MSE approximation during a model-based bit allocation process in Section 5, and conclude in Section 6.

2 PRELIMINARIES

As introduced previously, the final objective of our work is to design an efficient wavelet-based geometry coder for semiregular triangular meshes. For this purpose, we aim to develop a bit allocation process to optimize the trade off between the reconstructed MSE relative to the geometry quantization and the global bitrate.

2.1 Overall Coding Scheme

The overall scheme of the wavelet coder used in this paper is presented in Fig. 2. The source semiregular mesh M is transformed by a Discrete Wavelet Transform DWT in several subbands of wavelet coefficients. The coordinates of the wavelet coefficients are then encoded with scalar quantizers SQ (depending on the quantization steps computed during the allocation process). An entropy coding is finally applied on the quantized coefficients to obtain the bitstream. In parallel, the coarse mesh connectivity can be encoded with any connectivity coder. In this paper, we use the coder of [12].

2.2 The Allocation Process

The objective of the allocation process is to determine the set of the quantization steps $\{q\}$ (used to quantize the subbands) that minimizes the distortion defined by the MSE σ_ε^2 relative to the reconstructed mesh geometry, at one given specific bitrate R_{target} . This can be modeled by the following problem \mathcal{P} :

$$(\mathcal{P}) \begin{cases} \text{minimize} & \sigma_\varepsilon^2(\{q\}) \\ \text{with constraint} & R_T(\{q\}) = R_{target}, \end{cases} \quad (1)$$

where R_T represents the total bitrate, R_{target} a user-given bitrate, and $\{q\}$ the set of quantization steps.

This constrained allocation problem can be formulated by a Lagrangian criterion:

$$J_\lambda(\{q\}) = \sigma_\varepsilon^2(\{q\}) + \lambda(R_T(\{q\}) - R_{target}), \quad (2)$$

with λ the lagrangian operator. Hence, the solutions of the allocation problem \mathcal{P} , i.e., the optimal quantization steps $\{q^*\}$ are obtained by minimizing this lagrangian criterion. So, we have to solve the following system [6]:

$$\begin{cases} \frac{\partial J_\lambda(\{q\})}{\partial q} = 0 \\ \frac{\partial J_\lambda(\{q\})}{\partial \lambda} = 0. \end{cases} \quad (3)$$

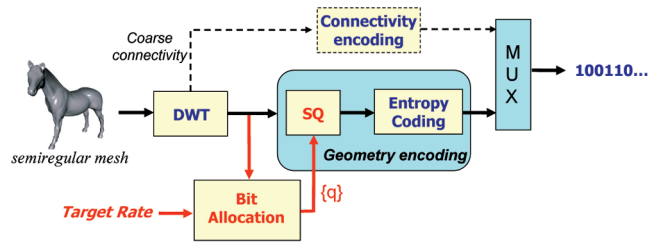


Fig. 2. Proposed geometry coder.

For more explanations about the resolution of this problem, see [6].

2.3 Problem Statement

Let us now introduce the problem statement relative to this allocation process. On one hand, the distortion relative to the reconstructed mesh is computed on the mesh geometry in the Euclidean space. On the other hand, the losses due to the quantization are relative to the subbands of wavelet coefficients. It means that, each time an estimation of the reconstructed MSE is required during the allocation process, we have to apply the synthesis filters of the wavelet transform on the quantized coefficients before the computation. This leads to a complex allocation process and, consequently, a slow compression algorithm. So, it should be more relevant to express the reconstructed MSE directly from the quantization errors of each coefficient subband, in order to reduce the algorithm complexity and to speed the process up.

This relation has already been developed in previous works [7], [8], [9], [10], [11]. It has indeed been shown that the MSE of a reconstructed signal can be approximated by a weighted sum of the MSE of each coefficient subband. This is due to the fact that using biorthogonal filters weights the amount of quantization error which appears on a reconstructed data. Among the previous works, Usevitch derived, for instance, the weighting of the quantization error in the specific case of dyadic filtering of images [11]. More generally, Park and Haddad [9] have defined this weighting for multidimensional signals across an m -channel wavelet coder. However, this approximation has been only computed on canonical sampling grids (in image or video coding, for instance) and never exploited, to our knowledge, in the domain of mesh processing.

Therefore, in the next section, we propose to derive the formulation of the weighted MSE for a triangular mesh geometry. Moreover, as our work is focused only on wavelet transforms based on lifting schemes, we attempt to express the weighting only in a function of the prediction and update operators (contrary to all the previous works which have formulated the weighting in function of the coefficients of the low pass and high pass synthesis filters).

3 MSE APPROXIMATION ON A TRIANGULAR EDGE LATTICE

Let us first introduce some global notations useful for the understanding of the rest of the paper.

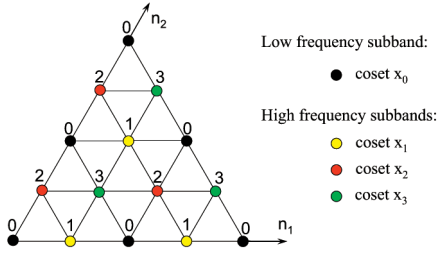


Fig. 3. Downsampling grid of the triangular edge lattice.

3.1 Notations

Let us define a sampled signal s as a sequence of real-valued numbers indexed by a finite set \mathcal{K} :

$$s = \{s(\mathbf{k}) \in \mathbb{R} \mid \mathbf{k} \in \mathcal{K}\}, \quad (4)$$

where $\mathcal{K} = \Gamma \mathbb{Z}^d$ with Γ an invertible $d \times d$ matrix permitting to obtain data sampled on lattices other than the canonical lattice \mathbb{Z}^d , for instance, the triangular edge lattice used in the rest of the paper.

A sublattice of \mathcal{K} can be obtained by $D\mathbb{Z}^d$, where D is a dilation matrix $d \times d$. The determinant of D is an integer $m \in \mathbb{Z}$. Then, the lattice \mathbb{Z}^d can be written as a sum of sublattices

$$\mathbb{Z}^d = \bigcup_{j=0}^{m-1} (D\mathbb{Z}^d + \mathbf{t}_j), \quad (5)$$

with $\mathbf{t}_j \in \mathbb{Z}^d$ the shift related to the j^{th} coset. Hence, we can define a coset s_i as the set of elements of the signal s corresponding to the sublattice $\mathcal{L} = D\mathbb{Z}^d + \mathbf{t}_i$, and given by

$$s_i(\mathcal{L}) = \{s(D\mathbf{k} + \mathbf{t}_i) \mid \mathbf{k} \in \mathbb{Z}^d\}. \quad (6)$$

Note that $s_i(\mathcal{L})$ is a sequence of real-valued numbers indexed by \mathbb{Z}^d and not by \mathcal{L} [13].

According to the definition of a sublattice, an m -channel filter bank $\{g_i\}$ on a lattice \mathcal{K} can be formulated according to the polyphase notation as:

$$G_i(\mathbf{z}) = \sum_{j=0}^{m-1} \mathbf{z}^{-\mathbf{t}_j} G_{i,j}(\mathbf{z}^D) \text{ for } i \in \{0, \dots, m-1\}, \quad (7)$$

with $G_{i,j}(\mathbf{z})$ the i, j th polyphase component of the synthesis filters, defined by

$$G_{i,j}(\mathbf{z}) = \sum_{\mathbf{k} \in \mathbb{Z}^d} g_i(D\mathbf{k} + \mathbf{t}_j) \mathbf{z}^{-\mathbf{k}}, \quad (8)$$

$\mathbf{z}^{-\mathbf{t}_j}$ the shift related to the j th coset given by

$$\mathbf{z}^{-\mathbf{t}_j} = \prod_{n=1}^d z_n^{-\mathbf{t}_j(n)}, \quad (9)$$

and

$$\mathbf{z}^D = \{\mathbf{z}^{\mathbf{d}_1}, \mathbf{z}^{\mathbf{d}_2}, \dots, \mathbf{z}^{\mathbf{d}_d}\}. \quad (10)$$

The vector \mathbf{d}_j is the j th column vector of the matrix D , and $\mathbf{z}^{\mathbf{d}_j}$ is given by:

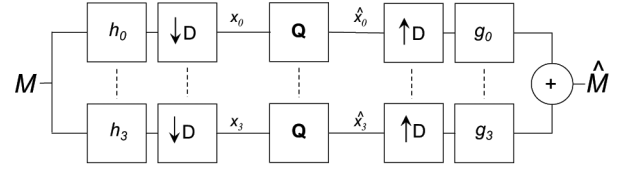


Fig. 4. Principle of a 4-channel wavelet coder.

$$\mathbf{z}^{\mathbf{d}_j} = \prod_{n=1}^d z_n^{\mathbf{d}_j(n)}. \quad (11)$$

3.2 Challenge

A semiregular mesh is based on a triangular edge lattice [13] (see Fig. 3). A wavelet transform for meshes corresponds consequently to a 4-channel filter bank. Hence, the geometry of a semiregular mesh M is transformed into four cosets $\{x_i, i = 0, \dots, 3\}$ on account of an analysis filter bank $\{h_i, i = 0, \dots, 3\}$ and a downsampling $\downarrow D$ (see Fig. 4).

The cosets are then quantized. Assuming that quantization error is an additive noise¹ [16], [15], the quantization error ε_i between the i th coset x_i and its quantized value \hat{x}_i is given by:

$$\varepsilon_i = (x_i - \hat{x}_i). \quad (12)$$

An upsampling $\uparrow D$ followed by a synthesis wavelet transform g_i provides the geometry of the reconstructed mesh \hat{M} . The challenge is thus to obtain the MSE relative to the reconstructed mesh geometry according to the quantization error of each subband and the knowledge of the synthesis filters.

3.3 MSE of the Quantized Mesh Geometry

In order to simplify the derivation, we propose to follow a deterministic approach, unlike the previous works of [7], [8], [9], [10], [11] which follow a statistical approach. So, let us consider the geometry of the source mesh M as a realization of a stationary and ergodic random process [16]. The total quantization error ε can thus be considered as a deterministic quantity defined by

$$\varepsilon = \{\varepsilon(\mathbf{k}) = (M(\mathbf{k}) - \hat{M}(\mathbf{k})) \in \mathbb{R} \mid \mathbf{k} \in \mathcal{K}\},$$

with \mathcal{K} being the sampling grid. \mathcal{K} is given by $\mathcal{K} = \Gamma \mathbb{Z}^2$ with Γ an invertible matrix permitting to obtain the data sampled on the triangular edge lattice instead of the canonical lattice \mathbb{Z}^2 . However, we can assume Γ is the identity, since this matrix influences only the choice of the neighbors for the filters [13]. The MSE relative to the reconstructed mesh geometry can be written as:

$$\sigma_\varepsilon^2 = \frac{1}{N_s} [r_\varepsilon(\mathbf{0})], \quad (13)$$

where $r_\varepsilon(\mathbf{t})$ is the autocorrelation function of ε , $\mathbf{0}$ is the null vector of dimension 2, and N_s is the number of samples of the input signal. The energy of the signal ε , denoted $r_\varepsilon(\mathbf{0})$, can be developed by using the deterministic correlation function

1. We prefer using the ‘‘additive noise’’ model because some assumptions relative to the ‘‘gain-plus-additive noise’’ model proposed by Jayant [14] and used by Park and Haddad have often been controversial [15].

$$R_{\mathcal{E}}(\mathbf{z}) = \mathcal{E}(\mathbf{z}) \mathcal{E}(\mathbf{z}^{-1}), \quad (14)$$

with $\mathcal{E}(\mathbf{z})$ the z -transform of the reconstruction error ε_r and $\mathbf{z} = (z_1, z_2)$. According to Fig. 4, $\mathcal{E}(\mathbf{z})$ can be formulated in a function of the error of each coset x_i [17]:

$$\mathcal{E}(\mathbf{z}) = \sum_{i=0}^3 G_i(\mathbf{z}) \mathcal{E}_i(\mathbf{z}^D), \quad (15)$$

with $\mathcal{E}_i(\mathbf{z})$ and $G_i(\mathbf{z})$, respectively, the z -transform of the reconstruction ε_i related to the coset x_i and of the synthesis filter g_i . D is the dilation matrix defined in Section 3.1. By assuming there is no cross-correlation between errors $\varepsilon_i(\mathbf{k})$ and $\varepsilon_i(\mathbf{k}')$ (for all $\mathbf{k} \neq \mathbf{k}'$) [16], [15], we can write

$$\mathcal{E}_i(\mathbf{z}^D) \mathcal{E}_j(\mathbf{z}^{-D}) = \delta_{i,j} R_{\mathcal{E}_i}(\mathbf{z}^D)$$

with $R_{\mathcal{E}_i}(\mathbf{z})$ the z -transform of the autocorrelation function of the reconstruction error ε_i , and $\delta_{i,j}$ the Krönercker symbol defined by

$$\delta_{i,j} = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if } i \neq j. \end{cases}$$

Hence, (14) and (15) provide:

$$R_{\mathcal{E}}(\mathbf{z}) = \sum_{i=0}^3 R_{G_i}(\mathbf{z}) R_{\mathcal{E}_i}(\mathbf{z}^D). \quad (16)$$

Applying the inverse z -transform on (16) yields the formulation of the autocorrelation function of the reconstruction error:

$$r_{\varepsilon}(\mathbf{t}) = \sum_{i=0}^3 \sum_{\tau} r_{g_i}(\tau) r_{\varepsilon_i}(D\mathbf{t} - \tau). \quad (17)$$

The energy $r_{\varepsilon}(\mathbf{0})$ of the signal ε is then given by:

$$r_{\varepsilon}(\mathbf{0}) = \sum_{i=0}^3 \sum_{\tau} r_{g_i}(\tau) r_{\varepsilon_i}(-\tau). \quad (18)$$

By assuming that the quantization error samplings are uncorrelated [16], $r_{\varepsilon_i}(-\tau) = 0$ if $\tau \neq \mathbf{0}$ and, consequently,

$$r_{\varepsilon}(\mathbf{0}) = \sum_{i=0}^3 r_{g_i}(\mathbf{0}) r_{\varepsilon_i}(\mathbf{0}). \quad (19)$$

Now, the problem is to deal with $r_{g_i}(\mathbf{0})$ and $r_{\varepsilon_i}(\mathbf{0})$. In the Appendix, we show that the energy $r_{g_i}(\mathbf{0})$ of the synthesis filter can be developed in

$$r_{g_i}(\mathbf{0}) = \sum_{j=0}^3 \sum_{\mathbf{k} \in \mathbb{Z}^2} g_i(D\mathbf{k} + \mathbf{t}_j)^2, \quad (20)$$

with $g_i(D\mathbf{k} + \mathbf{t}_j) = g_{i,j}(\mathbf{k})$ the coefficient k of the j th polyphase component of the synthesis filter i . In parallel, by assuming that the quantization error samplings are uncorrelated [16], the energy $r_{\varepsilon_i}(\mathbf{0})$ of the quantization error is:

$$r_{\varepsilon_i}(\mathbf{0}) = \sum_{\mathbf{k} \in \mathbb{Z}^2} \varepsilon_i(\mathbf{k})^2 = N_{s_i} \sigma_{\varepsilon_i}^2, \quad (21)$$

where $\sigma_{\varepsilon_i}^2$ stands for the MSE of the coset x_i and N_{s_i} the number of samples of x_i .

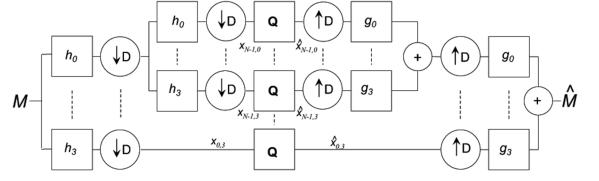


Fig. 5. A 4-channel wavelet coder with two levels of decomposition.

3.4 Solution for a One-Level Decomposition

Finally, by merging (20), (21), and (19) in (13), we obtain an expression of the MSE relative to the reconstructed mesh geometry:

$$\sigma_{\varepsilon}^2 = \sum_{i=0}^3 \frac{N_{s_i}}{N_s} w_i \sigma_{\varepsilon_i}^2 \quad \text{with } w_i = \sum_{j=0}^3 \sum_{\mathbf{k} \in \mathbb{Z}^2} g_{i,j}(\mathbf{k})^2, \quad (22)$$

where $g_{i,j}(\mathbf{k})$ represents the coefficient k of the j th polyphase component of the synthesis filter i . We stress that the weights w_i depend only on the polyphase matrix components of the synthesis filters. Now, in case of lifting schemes, the polyphase components depend only on the predictive and update operators (see the next section). We finally obtain an approximation specific to the lifting schemes, as expected.

3.5 Solution for an N -Level Decomposition

Wavelet coders generally exploit several levels of decomposition by applying several times the wavelet transform on the coset of lowest frequency. For example, the z -transform of the reconstruction error ε according to a two-level decomposition (see Fig. 5) can be written as:

$$\begin{aligned} \mathcal{E}(\mathbf{z}) &= G_0(\mathbf{z}) \cdot \sum_{l=0}^{M-1} G_l(\mathbf{z}^D) \mathcal{E}_{i,j}(\mathbf{z}^{D^2}) \\ &+ \sum_{l=1}^{M-1} G_l(\mathbf{z}) \mathcal{E}_{i,j}(\mathbf{z}^{D^2}), \end{aligned} \quad (23)$$

where $\mathcal{E}_{i,j}(\mathbf{z})$ stands for the z -transform of the quantization error $\varepsilon_{i,j}$ related to the coset (i, j) , with i the level of decomposition and j the channel index. By the same way as for the one-level decomposition, the MSE on a triangular edge lattice across a two-level wavelet coder can be simplified in:

$$\begin{aligned} \sigma_{\varepsilon}^2 &= \frac{N_{s_{0,0}}}{N_s} w_0 \cdot \sum_{l=0}^3 \left[\frac{N_{s_{1,l}}}{N_{s_{0,0}}} w_l \sigma_{\varepsilon_{1,l}}^2 \right] \\ &+ \sum_{l=1}^3 \left[\frac{N_{s_{0,l}}}{N_s} w_l \sigma_{\varepsilon_{0,l}}^2 \right], \end{aligned} \quad (24)$$

where $N_{s_{i,j}}$ is the number of samples of the i, j th coset. Thus, it is easy to generalize (24) to an N -level decomposition:

$$\sigma_{\varepsilon}^2 = \frac{N_{s_{N-1,0}}}{N_s} W_{N-1,0} \sigma_{\varepsilon_{N-1,0}}^2 + \sum_{i=0}^{N-1} \sum_{l=1}^3 \frac{N_{s_{i,l}}}{N_s} W_{i,l} \sigma_{\varepsilon_{i,l}}^2, \quad (25)$$

where $W_{i,l}$ represent the weights relative to the coset (i, j) , with i the level of decomposition and j the channel index.²

2. $(N-1)$ corresponds to the lowest decomposition level. Hence, the index $(N-1, 0)$ is relative to the low frequency subband of the mesh geometry.

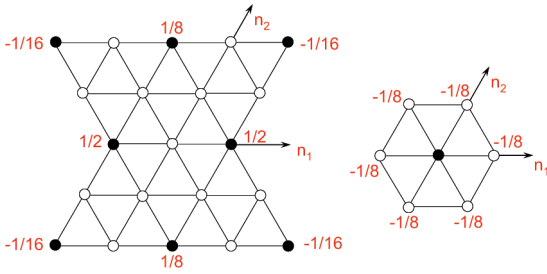


Fig. 6. Neighborhood used to design the prediction operator p_1 (left) and the update operator u_1 (right) of the lifted Butterfly scheme related to the coset x_1 . The operators related to the cosets x_2 and x_3 are obtained by the same way, but rotated according to their orientation.

These weights depend on the weights $\{w_i\}$ defined for a one-level decomposition. They are given by

$$W_{i,l} = (w_0)^i w_l. \quad (26)$$

4 COMPUTATION OF WEIGHTS FOR THE BUTTERFLY-BASED LIFTING SCHEME

In this paper, we only focus on the Butterfly-based lifting schemes [13] since our coding algorithm uses these filters. However, the proposed weighting can obviously be applied to any lifting scheme for semiregular meshes.³

There generally exist two different versions for the Butterfly-based lifting scheme: the *lifted* version (a prediction step and an update step) and the *unlifted* version (only a prediction step). The prediction and update operator filters of the lifted Butterfly scheme are presented in Fig. 6 [13]. The description of this lifting scheme can be found in [20].

As stated before, only the polyphase components are needed to compute the weights. We so consider the polyphase matrix of a 4-channel lifting scheme [13], given by

$$G = \begin{pmatrix} 1 & p_1 & p_2 & p_3 \\ -u_1 & 1 - u_1 p_1 & -u_1 p_2 & -u_1 p_3 \\ -u_2 & -u_2 p_1 & 1 - u_2 p_2 & -u_2 p_3 \\ -u_3 & -u_3 p_1 & -u_3 p_2 & 1 - u_3 p_3 \end{pmatrix}, \quad (27)$$

with p_i and u_i the prediction and update operators associated to the i th coset. Hence, identifying this matrix with the operators p_i and u_i of any 4-channel lifting scheme and using the formulation (22) allows to compute the corresponding weights w_i .

The weights for the *lifted* Butterfly-based transform, computed by substituting the z -transform of the prediction and update operators in each component of the polyphase matrix given by (27), are:

$$\begin{cases} w_0 = \frac{169}{256} \simeq 0.66015625 \\ w_1 = \frac{1727}{2048} \simeq 0.843261715 \\ w_2 = \frac{1727}{2048} \simeq 0.843261715 \\ w_3 = \frac{1727}{2048} \simeq 0.843261715. \end{cases} \quad (28)$$

3. After this work was finished, we noticed that two Loop-based lifting schemes were proposed in [18], [19]. The framework given here is sufficient to compute the corresponding weightings.

Similarly, the weights for the *unlifted* Butterfly-based transform are:

$$\begin{cases} w_0 = \frac{169}{256} \simeq 0.66015625 \\ w_1 = 1 \\ w_2 = 1 \\ w_3 = 1. \end{cases} \quad (29)$$

4.1 Complexity Reduction

As said previously, using this MSE approximation leads to faster distortion computations during the bit allocation since we do not have to apply the synthesis filters before. In the specific case of the Butterfly-based lifting scheme, the complexity is the following:

- Eight multiplications and eight additions per sample coordinate during the prediction stage, i.e., for the high frequency subbands x_1 , x_2 , and x_3 . In other words, 48 arithmetic operations (*a.o.*) per wavelet coefficient.
- Six multiplications and six additions per sample coordinate during the update stage, i.e., for the low frequency coset x_0 . In other words, 36 *a.o.* per low frequency coefficient.

Thus, the complexity $C_{lb}(N)$ for a N -level decomposition of the *lifted* version of the Butterfly-based scheme is

$$\sum_{i=0}^{N-1} \left[36 \times N_{s_{i,0}} + 36 \times \sum_{l=0}^{i-1} N_{s_{i,l}} + 48 \times \sum_{l=1}^3 N_{s_{i,l}} \right] a.o. \quad (30)$$

In parallel, the complexity $C_{ub}(N)$ for a N -level decomposition of the *unlifted* version of the butterfly-based scheme (only a prediction step) is

$$\sum_{i=0}^{N-1} \left[48 \times \sum_{l=1}^3 N_{s_{i,l}} \right] a.o. \quad (31)$$

As we use an iterative algorithm to solve the optimization problem (like in [6], for instance), the distortion is computed several times during the allocation process. The convergence of our algorithm is generally reached in approximately five iterations (the maximal number of iterations we observe is 10). Thus, given I the number of iterations, the whole complexity reduction relative to the use of the weighted MSE is given by $I \times C_{lb}(N)$ for the *lifted* version of the butterfly-based scheme, and by $I \times C_{ub}(N)$ for the *unlifted* one. For instance, the synthesis for the lifted version takes about 0.5 second on a Pentium III 1GHz with 512 *Mbytes*, involving a significant average time cost reduction of 2.5 seconds for the allocation process (for five iterations). Furthermore, as explained in [6], exploiting this MSE approximation allows us to develop a model-based algorithm for the allocation process as well. Combining the MSE approximation with a model-based algorithm finally leads to a more significant time cost reduction. As an example, the allocation is finally processed in less than 0.4 second for instance on a Pentium III 1GHz with 512 *Mbytes* RAM.

5 SIMULATION RESULTS

In order to evaluate the coding gain when using the weighted MSE as distortion criterion during the bit allocation, we experiment the compression algorithm presented in Section 2 (see Fig. 2) according to two cases:

- the distortion criterion is the MSE approximation with the weighting obtained in Section 3 and
- the distortion criterion is a so-called “naïve” weighted MSE. In that case, the weights depend only on the resolution levels and, thus, are defined by inverse powers of 2.

Moreover, to show that the weighting can be exploited for any kind of semiregular meshes (independently of the remeshing method) and for any scheme, we propose two versions of our coder:

- the first version, named the *MAPS coder*, deals with meshes issued from the remesher *MAPS* [21] and includes the *lifted* Butterfly scheme, and
- the second version, named the *NORMAL coder*, deals with *Normal Meshes* [22] and includes the *unlifted* Butterfly scheme.

Then, we compare the quality of the meshes obtained after decompression, quantized at the same given bitrate. Figs. 7a, 7b, 7c, and 7d show, respectively, the corresponding curve PSNR/bitrate for the models BUNNY, VENUS, RABBIT, and HORSE encoded with the proposed *MAPS coder*. As numerous papers about wavelet geometry coders [3], [4], [23], the PSNR is given by

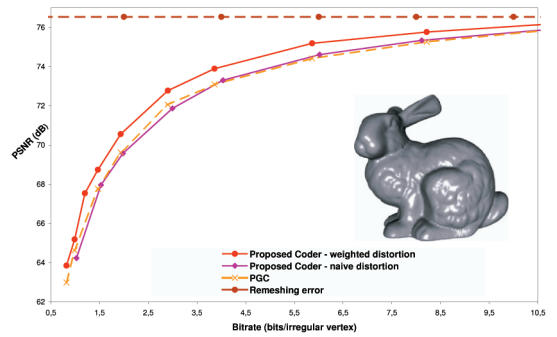
$$PSNR = 20 \log_{10} \left(\frac{\text{peak}}{d_s} \right),$$

where *peak* is the bounding box diagonal of the original object, and d_s is the Root MSE between the original irregular mesh and the reconstructed semiregular one (computed with *MESH* [24]). The bitrate is reported with respect to the number of vertices in the original irregular mesh (b/iv). In addition, Figs. 8a, 8b, 8c, and 8c show, respectively, the curves PSNR/bitrate for the models SKULL, HORSE, RABBIT, and VENUS, encoded with the proposed *NORMAL coder*.

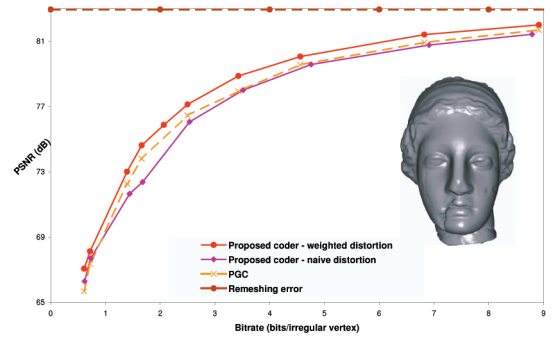
Theoretically the quantization error assumptions used during derivation in Section 3 are only valid for high bitrates [16], and we could expect that the coding gain is limited for low bitrates. However, compared to the “naïve” proposed coder, we globally observe that using the weighted MSE significantly improves the coding performances for any model, any version of the lifting scheme, and at any bitrate. Note that the gain reaches more than 3.4 dB on the *Normal mesh* SKULL and 2.7 dB on the *MAPS mesh* VENUS at low bitrates.

Also, Fig. 9 provides the visual benefits by showing the distribution of the local reconstruction error on several models. The colour corresponds to the magnitude of the distance point-surface (see Fig. 9a) between the original irregular mesh and the reconstructed semiregular one (computed with *MESH* [24]). We globally observe that, for one specific bitrate, the quality of the reconstructed meshes is always higher when the weighted MSE is exploited during the allocation process.

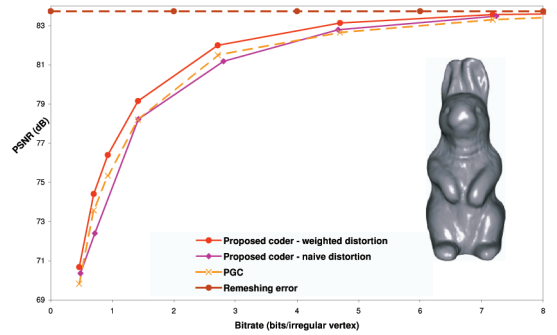
In order to evaluate the performances of our algorithm when using the proposed MSE approximation as distortion criterion of the bit allocation, we also compare our algorithms with some state-of-the-art coders. To be coherent, we obviously compare our *MAPS coder* with the zerotree coder for *MAPS* meshes named *PGC* [3], exploiting the *lifted* version of the Butterfly-based transform. On the other hand, we compare our *Normal coder* with the zerotree coder named for *Normal meshes* *NMC* [4] and with *EQMC* [23], both exploiting the *unlifted* version of the Butterfly-based transform. The curves PSNR/bitrate (see Fig. 7 and Fig. 8) show



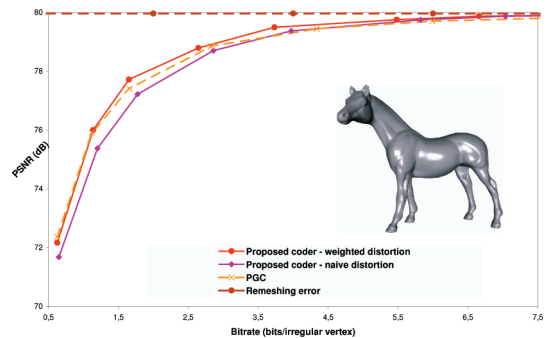
(a)



(b)



(c)



(d)

Fig. 7. PSNR curves for meshes encoded with the *MAPS coder* (*lifted* Butterfly-based transform). (a) BUNNY, (b) VENUS, (c) RABBIT, and (d) HORSE.

that, for *MAPS meshes*, the proposed coder provides better results (up to 3.5 dB) or, in the worst case, similar results. In parallel, our *Normal coder* always outperforms the state-of-the-art coders *NMC* and *EQMC* (up to more than 3.5 dB

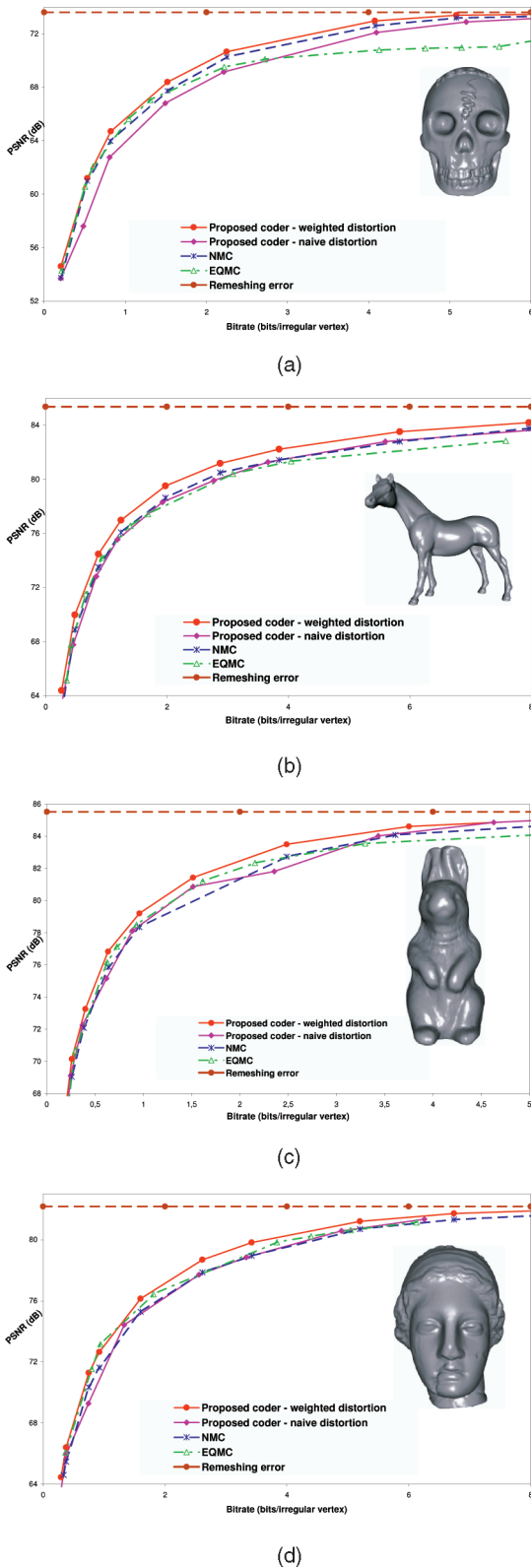


Fig. 8. PSNR curves for meshes encoded with the *NORMAL* coder (*unlifted* Butterfly-based transform). (a) SKULL, (b) HORSE, (c) RABBIT, and (d) VENUS.

compared to *NMC*). Notice that without the proposed weighting, our coders generally provide worse results than the state-of-the-art coders, showing definitively the interest of using the weighted MSE.

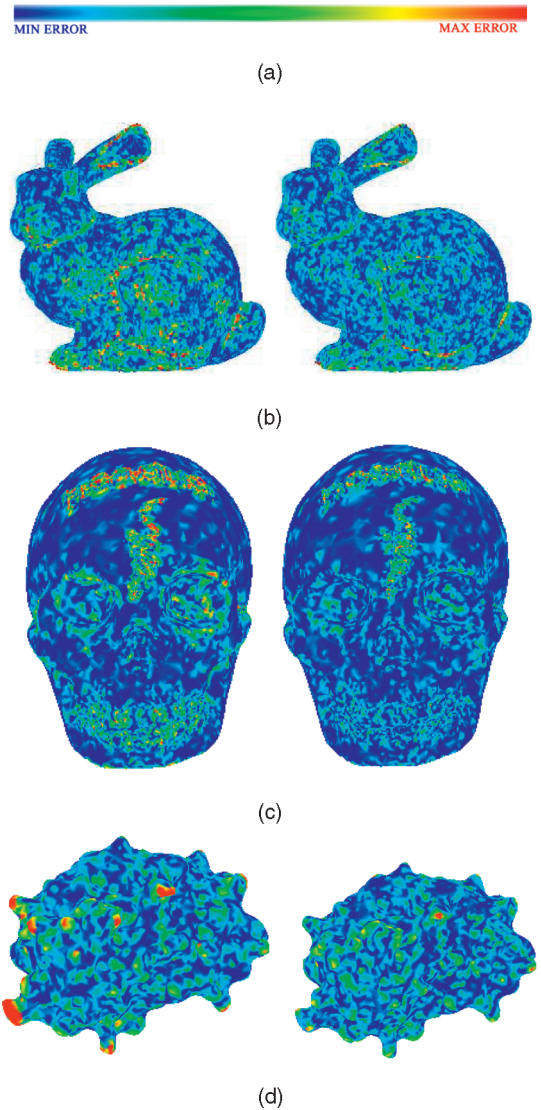


Fig. 9. Distribution of the reconstruction error on different models (at its finest resolution) according to the distortion criterion. (a) Error scale. The warmer colors correspond to the higher errors. (b) Naive MSE: 2.0 b/iv, PSNR = 69.58 dB. (c) Weighted MSE: 2.0 b/iv, PSNR = 70.92 dB. (d) Naive MSE: 2.2 b/iv, PSNR = 69.15 dB. (e) Weighted MSE: 2.2 b/iv, PSNR = 70.65 dB. (f) Naive MSE: 2.2 b/iv, PSNR = 54.9 dB. (g) Weighted MSE: 1.69 b/iv, PSNR = 57.54 dB.

6 CONCLUSIONS

In this paper, we proposed a weighted MSE relative to the triangular mesh geometry. The final objective was to propose an efficient model-based bit allocation process optimizing the performances of a wavelet-based geometry coder for semiregular meshes. We have particularly shown that the weights can depend only on the polyphase components of the synthesis filters, which is very useful in case of lifting schemes. Experimentally, we observe that using this MSE approximation as distortion criterion of a bit allocation process significantly improves the coding performances of a wavelet coder (for any kind of semiregular meshes, any bitrate, and any lifting scheme), so that the proposed algorithms outperform the relative state-of-the-art coders.

APPENDIX

Energy of the Synthesis Filter

The energy of the synthesis filter $r_{g_i}(\mathbf{0})$ is given by:

$$r_{g_i}(\mathbf{0}) = \frac{1}{2\pi j} \oint_{\Gamma} G_i(\mathbf{z}) G_i(\mathbf{z}^{-1}) \mathbf{z}^{-1} d\mathbf{z}. \quad (32)$$

According to (7), a synthesis filter bank $\{g_i\}$ on a triangular edge lattice can be formulated according to the polyphase notation:

$$G_i(\mathbf{z}) = \sum_{j=0}^3 \mathbf{z}^{-\mathbf{t}_j} G_{i,j}(\mathbf{z}^D) \text{ for } i \in \{0, \dots, 3\}, \quad (33)$$

with $G_{i,j}(\mathbf{z})$ the i, j th polyphase component of the synthesis filters, defined by

$$G_{i,j}(\mathbf{z}) = \sum_{\mathbf{k} \in \mathbb{Z}^2} g_i(D\mathbf{k} + \mathbf{t}_j) \mathbf{z}^{-\mathbf{k}}, \quad (34)$$

and $\mathbf{z}^{-\mathbf{t}_j}$ the shift relative to the j th coset. By exploiting (33) and (34), (32) can be developed in:

$$r_{g_i}(\mathbf{0}) = \frac{1}{2\pi j} \sum_{u=0}^3 \sum_{v=0}^3 \sum_{\mathbf{k} \in \mathbb{Z}^2} \sum_{\mathbf{k}' \in \mathbb{Z}^2} g_i(D\mathbf{k} + \mathbf{t}_u) g_i(D\mathbf{k}' + \mathbf{t}_v) \oint_{\Gamma} \mathbf{z}^{(-D\mathbf{k} + D\mathbf{k}' - \mathbf{t}_u + \mathbf{t}_v - 1)} d\mathbf{z}. \quad (35)$$

Using the Cauchy theorem, that is,

$$\frac{1}{2\pi j} \oint_{\Gamma} \mathbf{z}^{\mathbf{l}-1} d\mathbf{z} = \begin{cases} 1 & \text{if } \mathbf{l} = \mathbf{0}, \\ 0 & \text{else,} \end{cases}$$

the integral operator of (35) is equal to $\mathbf{1}$ if

$$-D\mathbf{k} + D\mathbf{k}' - \mathbf{t}_u + \mathbf{t}_v = \mathbf{0} \quad (36)$$

is satisfied. The dilation matrix D being invertible, this condition becomes $(-\mathbf{k} + \mathbf{k}') - (D^{-1}\mathbf{t}_u - D^{-1}\mathbf{t}_v) = \mathbf{0}$. From [13], we know that $D^{-1}\mathbf{t}_j$ is restricted to the unit hypercube, that is, $[0, 1)^2$. On the other hand, $\mathbf{k} \in \mathbb{Z}^2$. These two remarks yield that

$$\begin{cases} (-\mathbf{k} + \mathbf{k}') & \in \mathbb{Z}^2 \\ (D^{-1}\mathbf{t}_u - D^{-1}\mathbf{t}_v) & \in (-1, 1)^2. \end{cases} \quad (37)$$

As $\mathbb{Z}^d \cap (-1, 1)^2 = \mathbf{0}$, we have to solve separately $(-\mathbf{k} + \mathbf{k}') = \mathbf{0}$ and $(D^{-1}\mathbf{t}_u - D^{-1}\mathbf{t}_v) = \mathbf{0}$ to satisfy (36). Consequently, the set of solutions is $\mathbf{k} = \mathbf{k}'$ and $\mathbf{u} = \mathbf{v}$. Finally, the energy of the synthesis filter $r_{g_i}(\mathbf{0})$ is given by

$$r_{g_i}(\mathbf{0}) = \sum_{j=0}^3 \sum_{\mathbf{k} \in \mathbb{Z}^2} g_i(D\mathbf{k} + \mathbf{t}_j)^2, \quad (38)$$

with $g_i(D\mathbf{k} + \mathbf{t}_j) = g_{i,j}(\mathbf{k})$ the coefficient \mathbf{k} of the j th polyphase component of the synthesis filter i .

ACKNOWLEDGMENTS

Models are courtesy of Cyberware, Headus, The Scripps Research Institute, Washington University, and Stanford University. The authors are particularly grateful to Igor Guskov for providing them with his *Normal meshes*, Aaron Lee for providing them with his *MAPS meshes*, and Andrei Khodakovsky and Shridar Lavu for providing them with

executables of their mesh compression algorithms. They also want to acknowledge the anonymous reviewers for their advice which permitted them to improve the quality of this paper.

REFERENCES

- [1] "Information Technology—JPEG 2000, Image Coding System—Part 1: Core Coding System," ISO/IEC 15444-1, 2000.
- [2] M. Lounsbury, T. DeRose, and J. Warren, "Multiresolution Analysis for Surfaces of Arbitrary Topological Type," *Trans. Graphics*, vol. 16, no. 1, 1997.
- [3] A. Khodakovsky, P. Schröder, and W. Sweldens, "Progressive Geometry Compression," *Proc. SIGGRAPH*, 2000.
- [4] A. Khodakovsky and I. Guskov, "Normal Mesh Compression," *Geometric Modeling for Scientific Visualization*, Springer-Verlag, 2002.
- [5] F. Payan and M. Antonini, "MSE Approximation for Model-Based Compression of Multiresolution Semiregular Meshes," *Proc. IEEE EUSIPCO 2005 (13th European Conf. Signal Processing)*, Sept. 2005.
- [6] F. Payan and M. Antonini, "An Efficient Bit Allocation for Compressing Normal Meshes with an Error-Driven Quantization," *Computer Aided Geometric Design*, special issue on geometric mesh processing, vol. 22, pp. 466-486, July 2005.
- [7] J. Woods and T. Naveen, "A Filter Based Bit Allocation Scheme for Subband Compression of HDTV," *IEEE Trans. Image Processing*, 1992.
- [8] C. Cheong, K. Aizawa, T. Saito, and M. Hatori, "Subband Image Coding with Biorthogonal Wavelets," *IEICE Trans. Fundamentals*, vol. E75-A, no. 7, July 1992.
- [9] K. Park and R.A. Haddad, "Modeling, Analysis, and Optimum Design of Quantized M-Band Filter Banks," *IEEE Trans. Signal Processing*, vol. 43, no. 11, Nov. 1995.
- [10] P. Moulin, "A Multiscale Relaxation Algorithm for SNR Maximization in Nonorthogonal Subband Coding," *IEEE Trans. Image Processing*, vol. 4, no. 9, 1995.
- [11] B. Usevitch, "Optimal Bit Allocation for Biorthogonal Wavelet Coding," *Proc. IEEE Data Compression Conf.*, Apr. 1996.
- [12] C. Touma and C. Gotsman, "Triangle Mesh Compression," *Proc. Conf. Graphics Interface '98*, pp. 26-34, 1998.
- [13] J. Kovacevic and W. Sweldens, "Wavelet Families of Increasing Order in Arbitrary Dimensions," *IEEE Trans. Image Processing*, 1999.
- [14] N.S. Jayant and P. Noll, *Digital Coding of Waveforms*, E. Cliffs, ed., New Jersey: Prentice Hall, 1984.
- [15] R. Gray and T. Stockham, "Dithered Quantizers," *IEEE Trans. Information Theory*, vol. 39, no. 3, May 1993.
- [16] A. Gersho and R. Gray, *Vector Quantization and Signal Compression*. Norwell, Mass.: Kluwer Academic Publishers, 1992.
- [17] M. Vetterli and J. Kovacevic, *Wavelets and Subband Coding*. Engelwood Cliffs, N.J.: Prentice Hall PTR, 1995.
- [18] D. Li, K. Qin, and H. Sun, "Unlifted Loop Subdivision Wavelets," *Proc. 12th Pacific Conf. Computer Graphics and Applications*, 2004.
- [19] M. Bertram, "Biorthogonal Loop-Subdivision Wavelets," *Computing*, vol. 72, nos. 1-2, pp. 29-39, 2004.
- [20] P. Schröder and W. Sweldens, "Spherical Wavelets: Efficiently Representing Functions on the Sphere," *Proc. SIGGRAPH '95*, pp. 161-172, 1995.
- [21] A. Lee, W. Sweldens, P. Schröder, P. Cowsar, and D. Dobkin, "MAPS: Multiresolution Adaptive Parametrization of Surfaces," *Proc. SIGGRAPH*, 1998.
- [22] I. Guskov, K. Vidimce, W. Sweldens, and P. Schröder, "Normal Meshes," *Siggraph 2000, Computer Graphics Proc.*, pp. 95-102, 2000.
- [23] S. Lavu, H. Choi, and R. Baraniuk, "Geometry Compression of Normal Meshes Using Rate-Distortion Algorithms," *Proc. Eurographics/ACM SIGGRAPH Symp. Geometry Processing*, 2003.
- [24] N. Aspert, D. Santa-Cruz, and T. Ebrahimi, "Mesh: Measuring Errors between Surfaces Using the Hausdorff Distance," *Proc. IEEE Int'l Conf. Multimedia and Expo*, vol. 1, pp. 705-708, 2002.



Frédéric Payan received the MS degree in image processing and computer science in 1999 and the PhD degree in signal and image processing in December 2004 from the University of Nice-Sophia Antipolis (France). The work for his PhD was supported by a grant from the region Provence Alpes Côte d'Azur and Opteway Corporation in Sophia Antipolis (France). He was a research and teaching assistant at the University of Nice-Sophia Antipolis until August

2005 and is currently a postdoctoral research fellow until August 2006 with the Geometric Modeling Group of the Laboratory LMC-IMAG (CNRS) at Grenoble (France). His research interests include signal processing and, particularly, mesh and image coding, multiresolution analysis, information theory, wavelet transform, geometry compression, and computational geometry. He is currently working on animated mesh compression and multiresolution analysis for deforming surfaces.



Marc Antonini received the PhD degree in electrical engineering from the University of Nice-Sophia Antipolis (France) in 1991 and the "Habilitation à Diriger des Recherches" from the University of Nice-Sophia Antipolis (France) in 2003. He was a postdoctoral fellow at the Centre National d'Etudes Spatiales (CNES, Toulouse, France), in 1991 and 1992. He joined the CNRS in 1993. He has been a research director at the I3S Laboratory and the University of Nice-

Sophia Antipolis since 2004. He is a regular reviewer for several journals (*IEEE Transactions on Image Processing*, *Information Theory and Signal Processing*, *IEE Electronics Letters*) and participated in the organization of the IEEE Workshop on Multimedia and Signal Processing 2001 in Cannes (France). He also participates in several national research and development projects with French industries and in several international academic collaborations. His research interests include multidimensional image processing, wavelet analysis, lattice vector quantization, information theory, still image and video coding, joint source/channel coding, inverse problem for decoding, multispectral image coding, multiresolution 3D mesh coding, and animated mesh compression.

► **For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/publications/dlib.**