Available online at www.sciencedirect.com
SCIENCE DIRECT®
Computer Aided Geometric Design 22 (2005) 466–486
COMPUTER AIDED GEOMETRIC DESIGN
www.elsevier.com/locate/cagd

An efficient bit allocation for compressing normal meshes with an error-driven quantization

Frédéric Payan *, Marc Antonini

Laboratoire I3S, UMR 6070 CNRS, Université de Nice-Sophia Antipolis, 2000 route des Lucioles, F-06903 Sophia Antipolis, France

Available online 23 May 2005

Abstract

We propose a new wavelet compression algorithm based on the rate-distortion optimization for densely sampled triangular meshes. Exploiting the *normal remesher* of Guskov et al., the proposed algorithm includes a wavelet transform and an original bit allocation optimizing the quantization of the wavelet coefficients. The allocation process minimizes the reconstruction error for a given bit budget. As distortion measure, we use the mean square error of the normal mesh quantization, expressed according to the quantization error of each subband. We show that this metric is a suitable criterion to evaluate the reconstruction error, i.e., the geometric distance between the input mesh and the quantized *normal* one. Moreover, to design a fast bit allocation, we propose a model-based approach, depending on distribution of the wavelet coefficients. Compared to the state-of-the-art methods for normal meshes, our algorithm provides improvements in coding performance, up to +2.5 dB compared to the original zerotree coder.
© 2005 Elsevier B.V. All rights reserved.

Keywords: Shape compression; Geometry coding; Normal meshes; Model-based bit allocation; Wavelet transform; Rate-distortion optimization; Multiresolution analysis

1. Introduction

Today triangular meshes can be defined by several millions of vertices, and more (Levoy, 1999). A simple representation of these highly detailed meshes is consequently huge. The compression is a relevant solution to allow a compact storage or a fast transmission in bandwidth-limited applications of these data.

* Corresponding author. Phone: +33 (0)4 92 94 27 22; fax: +33 (0)4 92 94 28 98.
E-mail addresses: fpayan@i3s.unice.fr (F. Payan), am@i3s.unice.fr (M. Antonini).

0167-8396/$ – see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.cagd.2005.04.001

Currently, more and more works consider the original mesh to be just one instance of the surface geometry. In that case, we talk about *shape compression* instead of *mesh compression* (Alliez and Gotsman, 2003). The *shape compression* considers the geometry to be the most important component of a mesh. Therefore this kind of algorithm generally tends to reduce the connectivity information to the minimum by remeshing the irregular input mesh with a *semi-regular remesher* (Lee et al., 1998; Guskov et al., 2000; Gu et al., 2002; Lee et al., 2000).

Among the existing schemes of semi-regular remeshing, the *normal meshes* (Guskov et al., 2000) are attractive for wavelet coding, particularly with the *unlifted butterfly* wavelet transform (Khodakovsky and Guskov, 2002). When the coefficients are expressed according to a system of local frames depending on the coarser mesh, the majority of coefficients have indeed no tangential component, and consequently almost all the geometry information lies in the normal components (Khodakovsky and Guskov, 2002). Hence, most of coefficients can be represented by a single scalar, instead of a three-dimensional vector like in (Khodakovsky et al., 2000) for instance. Therefore several wavelet coders exploit the normal meshes combined with the *unlifted butterfly* wavelet transform. Let us cite for example the coder NMC proposed by Khodakovsky and Guskov (2002). This coder is based on a zerotree coder developed in (Khodakovsky et al., 2000). The wavelet coefficients are first organized in a multiscale quadtree structure. Then, a zerotree coder (followed by an entropy coding) is applied on each component of the wavelet coefficients separately. Sim et al. (2002) proposed a progressive compression and an interactive transmission algorithm for normal meshes based on rate-distortion optimization. We can also cite the works of Lavu et al. (2003). The resulting compression algorithm EQMC is based on an *Estimation-Quantization* framework initially developed for 2D images (Lopresto et al., 1997). This algorithm exploits the spatial and inter-scale correlations of the normal meshes. The authors propose to find the best quantizer for each normal component depending on the normal components previously encoded, in the local neighborhood. This allows to optimize locally the trade-off between the bitrate and the quantization error, providing 0.5–1 dB improvement in coding performance compared to NMC (Khodakovsky and Guskov, 2002). In the same way, we proposed in previous works (2003) a bit allocation controlling the quantization error energy to dispatch the bits across wavelet subbands (Payan and Antonini, 2003a, 2003b).

The basic idea of these works is to optimize the trade-off between the bitrate and the quality of the reconstructed mesh either by minimizing the losses due to the geometry coding, or by reducing the bit budget. This principle called *bit allocation*, allows to improve the coding performances when a multiresolution analysis is performed.[1]

In order to improve the coding performances compared to the state-of-the-art coders for normal meshes *at a specific bitrate*, we propose in this paper a bit allocation process that optimizes the quantization of the *normal mesh* wavelet coefficients at a given bitrate R_{target}. We focus on the normal meshes because of its simple and multiscale representation, allowing an efficient multiresolution analysis and adaptive displaying according to the level-of-detail requirements or hardware capabilities (Khodakovsky and Guskov, 2002; Khodakovsky et al., 2000; Sim et al., 2002; Lavu et al., 2003). Precisely, we aim to find the best quantizer for each component subband such that the reconstruction error is minimized for the given

[1] Note that bit allocation is not only used in case of wavelet coding. Let us cite for instance Chow (1997) or Li et al. (1997) who proposed a coder allowing different regions of a mesh to be compressed with different precision in function on the level of details. King and Rossignac (1999) focused on the problem of balancing two forms of lossy mesh compression: reduction of the number of vertices by simplification techniques, and reduction of the bitrate per vertex coordinate. More recently, Karni and Gotsman (2000) proposed to truncate spectral coefficients according to a maximum RMS value given as input parameter.

bitrate R_{target}. A distortion measure is consequently needed to evaluate the reconstruction error of the quantized mesh.

Several distortion measures have been exploited for compression of irregular meshes (King and Rossignac, 1999; Karni and Gotsman, 2000; Sorkine et al., 2003; Luebcke and Halle, 2001). For instance, Karni and Gotsman (2000) introduce a metric which captures the visual difference between the original mesh and its approximation. Their criterion depends on the geometric distance and the *laplacian difference* between models. Unfortunately, we cannot use such a *vertex-to-vertex* measure since the proposed coder includes a remeshing technique modifying the topology of the input mesh. In that case, the widely used metric is the symmetric root mean square error between two surfaces (Cignoni et al., 1998), because it does not depend on the mesh sampling, or its connectivity. We refer to this error as the *surface-to-surface* (*S2S*) *distance*. A real computation of the S2S distance is a computationally intensive process. To overcome this problem, we argue that the mean square error relative to the normal mesh quantization, expressed according to the quantization error of each subband, is a suitable criterion to evaluate the reconstruction error between the input mesh and the quantized normal one. Furthermore, this criterion allows to use theoretical models for the bitrate and the distortion of each wavelet subband, involving a fast model-based algorithm of low computational complexity.

We finally design a wavelet coder that includes a bit allocation dispatching a given bit budget across the wavelet subbands according to their influence on the reconstructed mesh quality. Compared to the state-of-the-art coders for normal meshes (Khodakovsky and Guskov, 2002; Lavu et al., 2003), our compression algorithm provides performance gains, up to $+2.5$ dB compared to the original zerotree coder.

This paper is organized as follows. In Section 2, we introduce some background and notations on triangular meshes and briefly describe the normal meshes. In Section 3, we present our framework and the proposed compression algorithm. Then, we deal with a suitable distortion criterion to evaluate the reconstruction error of normal meshes in Section 4, and across a wavelet coder in Section 5. In Section 6 we introduce the proposed bit allocation and develop the model-based algorithm in Section 7. Finally, we give some experimental results in Section 8, and conclude in Section 9.

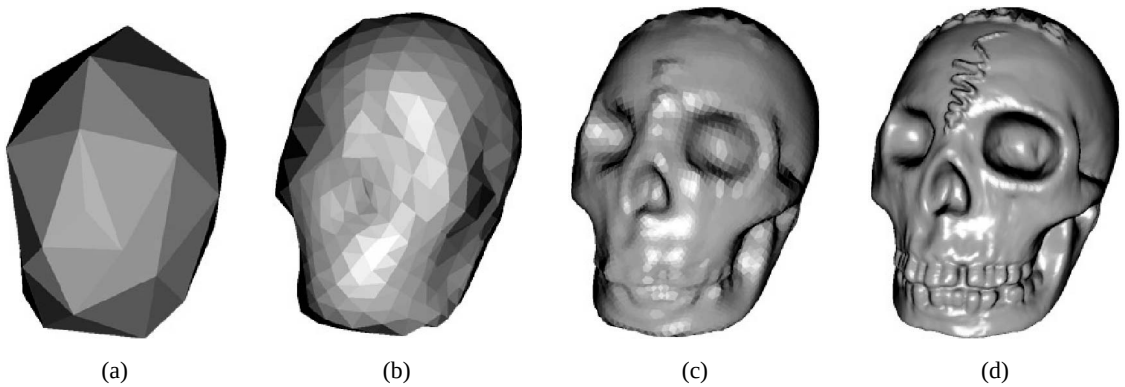
2. Background and notations

Triangular mesh. Let us denote a triangular mesh $\mathcal{M}$ as a pair $(\mathcal{V}, \mathcal{T})$, where $\mathcal{V}$ is a set of vertices defined by $\mathcal{V} = \{v_i = (v_i^x, v_i^y, v_i^z) \in \mathbb{R}^3 \mid 1 \leqslant i \leqslant |\mathcal{V}|\}$ with $|\mathcal{V}|$ the number of vertices, and $\mathcal{T}$ a set of triangular faces.

Semi-regular mesh. A semi-regular triangular mesh $\mathcal{M}_{sr}$ is a multiscale mesh, built by repeated regular subdivision of a base mesh $\mathcal{M}_0 = (\mathcal{V}_0, \mathcal{T}_0)$ (a coarse version of the original irregular mesh obtained by a simplification technique (Gotsman et al., 2002)), providing several meshes $\mathcal{M}_i$ ($\mathcal{M}_1 = (\mathcal{V}_1, \mathcal{T}_1)$, $\mathcal{M}_2 = (\mathcal{V}_2, \mathcal{T}_2), \ldots$) until the finest semi-regular mesh $\mathcal{M}_{sr} = (\mathcal{V}_{sr}, \mathcal{T}_{sr})$. These meshes have the notable property:

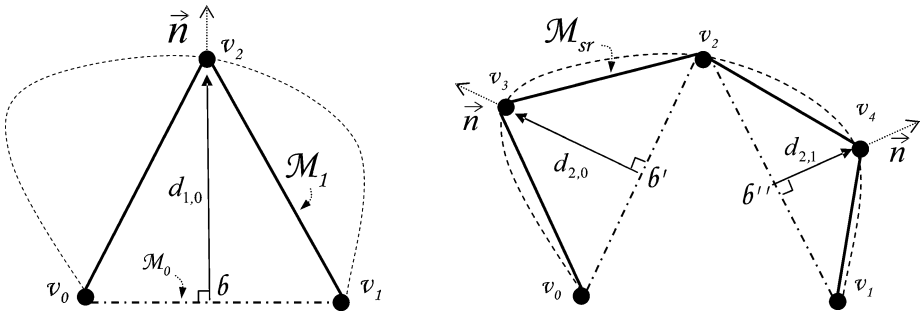
$$\mathcal{V}_0 \subset \mathcal{V}_1 \subset \cdots \subset \mathcal{V}_{sr}.$$

Fig. 1 shows an example of a semi-regular mesh at different resolution levels. The vertices added to obtain a finer mesh can be defined by a set of three-dimensional detail vectors $\mathcal{D}_i = \{d_{i,j} = (d_{i,j}^x, d_{i,j}^y, d_{i,j}^z) \in \mathbb{R}^3 \mid 1 \leqslant j \leqslant |\mathcal{D}_i|\}$, with $|\mathcal{D}_i|$ the number of details at the resolution level i. The set of details $\mathcal{D}_i$ describes the deformations between the mesh $\mathcal{M}_{i-1}$ and $\mathcal{M}_i$. The details are mostly computed in a local frame (Zorin



(a) (b) (c) (d)

Fig. 1. Multiresolution semi-regular version of SKULL at different levels of resolution, from level 2 (64 triangles) to level 8 (finest mesh with 262144 triangles). (a) Level 2; (b) level 4; (c) level 6; (d) finest mesh.



(a) (b)

Fig. 2. A *normal mesh* $\mathcal{M}_{sr}$ is obtained by successive connectivity subdivision of a coarse mesh $\mathcal{M}_0$. The detail vectors depend on the normals at the surface. (a) A finer mesh $\mathcal{M}_1$ is obtained from the coarse mesh $\mathcal{M}_0$ and a detail $d_{1,0}$. (b) The finest mesh $\mathcal{M}_{sr}$ is obtained from $\mathcal{M}_1$ and the details $d_{2,0}$ and $d_{2,1}$.

et al., 1997) induced by the tangent plane and the normal direction at the surface defined by the mesh of lower resolution (Khodakovsky et al., 2000). This means that we can distinguish the so-called *tangential components* from the *normal components* of detail vectors $d_{i,j}$:

- the *tangential components* are the coordinates $d^x_{i,j}$ and $d^y_{i,j}$ of detail vectors;
- the *normal components* are the coordinates $d^z_{i,j}$ of detail vectors.

Normal mesh. The normal meshes are attractive because majority of the details may be represented with a single number instead of a three-dimensional vector like in (Lee et al., 1998). These multiresolution meshes have the property that the details almost always lie in a known normal direction (see Fig. 2) (Guskov et al., 2000). This means that the tangential components tends to be equal to zero. This is currently the most compact representation of semi-regular meshes.

Quantized mesh. Let us denote a *quantized normal mesh* $\widehat{\mathcal{M}}_{sr}$ as a pair $\widehat{\mathcal{M}}_{sr} = (\widehat{\mathcal{V}}_{sr}, \mathcal{T}_{sr})$. $\widehat{\mathcal{V}}_{sr}$ represents the set of quantized vertices defined by $\widehat{\mathcal{V}}_{sr} = \{\hat{v}_i \in \mathbb{R}^3 \mid 1 \leqslant i \leqslant |\widehat{\mathcal{V}}_{sr}|\}$, where $\hat{v}_i = Q(v_i)$. $Q(.)$ is called the quantization operator associated to a quantization step q.

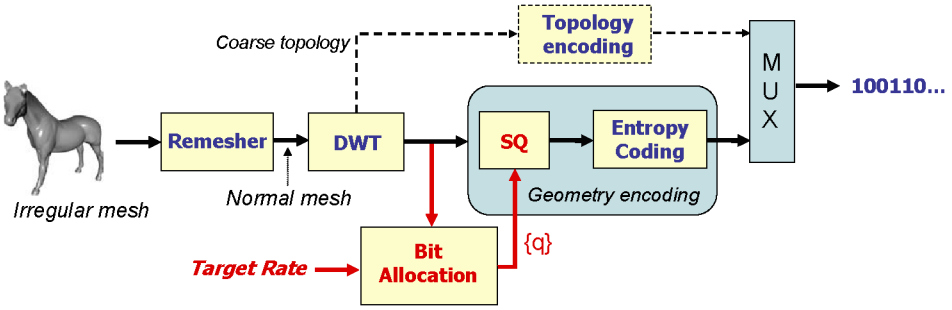


Fig. 3. Proposed geometry coder.

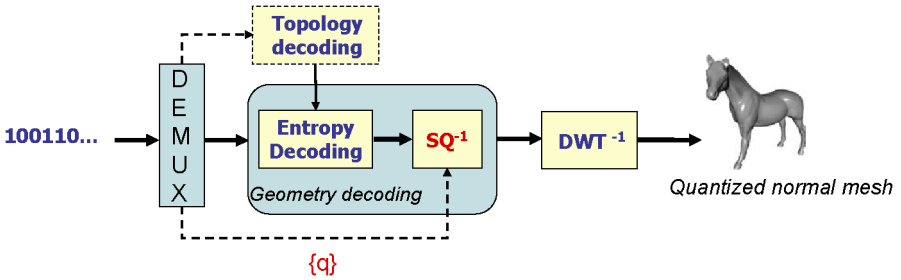


Fig. 4. Proposed geometry decoder.

3. Overview of the proposed approach

Figs. 3 and 4 present the global scheme of the proposed coder/decoder. The algorithm principle is described hereinafter. The normal remesher provides a semi-regular mesh $\mathcal{M}_{sr}$, from the irregular input one $\mathcal{M}_{ir}$. A N-level unlifted butterfly wavelet transform (Sweldens, 1998; Schröder and Sweldens, 1995) is then applied to obtain N subbands of three-dimensional wavelet coefficients. Using this wavelet transform ensures that wavelet coefficients remain in the normal direction (Khodakovsky and Guskov, 2002).

The *tangential* and *normal sets* (see Section 2) of wavelet coefficients are then encoded separately using uniform scalar quantizers SQ depending on the optimal quantization steps computed during the allocation process. An entropy coder adapted to the multiresolution semi-regular mesh (Payan and Antonini, 2003b) is finally applied. In parallel, the connectivity of the coarse mesh can be encoded with any topological coder. In this paper, we choose the efficient coder of Touma and Gotsman (1998). Finally, the two bitstreams are merged for transmission.

The goal of this paper is to propose a coder/decoder including a bit allocation process that optimizes the quality of the *quantized normal mesh*. A suitable distortion measure D_T is thus needed to evaluate the reconstruction error during the geometry encoding. In the next section, we deal with the choice of the distortion measure.

4. Choice of the distortion measure

4.1. The S2S distance as quality criterion

Several distortion measures have been exploited by single-rate mesh coders (King and Rossignac, 1999; Karni and Gotsman, 2000; Sorkine et al., 2003; Luebcke and Halle, 2001). In this paper, we choose as reconstruction error the symmetric root mean square error between two surfaces (Cignoni et al., 1998) also called the S2S distance. We choose this distance because it is generally used to evaluate the performances of coders based on remeshing (Khodakovsky et al., 2000; Khodakovsky and Guskov, 2002; Sim et al., 2002; Lavu et al., 2003). Indeed, this distance does not depend on the mesh sampling, or connectivity.

The distortion measure D_T is defined as the energy of the S2S distance between the irregular input mesh $\mathcal{M}_{ir}$ and the *quantized normal mesh* $\widehat{\mathcal{M}}_{sr}$:

$$D_T = d_S(\mathcal{M}_{ir}, \widehat{\mathcal{M}}_{sr})^2, \tag{1}$$

where $d_S(.,.)$ represents the S2S distance.

4.2. Definition of the S2S distance

The S2S distance between the two meshes $\mathcal{M}_{ir}$ and $\widehat{\mathcal{M}}_{sr}$ (Cignoni et al., 1998) is defined by

$$d_S(\mathcal{M}_{ir}, \widehat{\mathcal{M}}_{sr}) = \max\bigl[\bar{d}(\mathcal{M}_{ir}, \widehat{\mathcal{M}}_{sr}); \bar{d}(\widehat{\mathcal{M}}_{sr}, \mathcal{M}_{ir})\bigr], \tag{2}$$

where $\bar{d}(\mathcal{M}, \mathcal{M}')$ is the *unilateral distance* between two meshes (Cignoni et al., 1998), given by

$$\bar{d}(\mathcal{M}, \mathcal{M}') = \left(\frac{1}{|\mathcal{M}|} \iint\limits_{p\in\mathcal{M}} d(p, \mathcal{M}')^2 \,\mathrm{d}\mathcal{M}\right)^{1/2}. \tag{3}$$

$|\mathcal{M}|$ represents the area of $\mathcal{M}$, and $d(p, \mathcal{M}')$ represents the distance between a point p belonging to a surface represented by a mesh $\mathcal{M}$ and the surface represented by a mesh $\mathcal{M}'$. This distance is defined by

$$d(p, \mathcal{M}') = \min_{p'\in\mathcal{M}'} \|p - p'\|_2 = \bigl\|p - \mathrm{Proj}_{\mathcal{M}'}(p)\bigr\|_2, \tag{4}$$

with $\|.\|_2$ the L_2-norm, and $\mathrm{Proj}_{\mathcal{M}'}(p)$ the orthogonal projection of p over $\mathcal{M}'$.

A real computation of the S2S distance is a computationally intensive process. Moreover, this distance would be hard to optimize during the allocation process. To overcome this problem, we propose to use a simpler but suitable criterion to evaluate the reconstruction error. To this purpose, we make several assumptions.

4.3. First assumption: an optimal remeshing

Notice that the normal remesher provides a *normal mesh* $\mathcal{M}_{sr}$ very close to the original irregular mesh $\mathcal{M}_{ir}$. Table 1 shows that the S2S distance between these two meshes is negligible (lower than 0.016% of the bounding box diagonal).

Eq. (1) can then be approximated by

$$D_T \simeq d_S(\mathcal{M}_{sr}, \widehat{\mathcal{M}}_{sr})^2 = \max\bigl[\bar{d}(\mathcal{M}_{sr}, \widehat{\mathcal{M}}_{sr})^2; \bar{d}(\widehat{\mathcal{M}}_{sr}, \mathcal{M}_{sr})^2\bigr]. \tag{5}$$

Table 1
Remeshing error between the irregular mesh and the normal mesh (S2S distance relative to the bounding box diagonal). "Base" represents the number of triangles of the base mesh

Model	$\lvert\mathcal{V}_{ir}\rvert$	$\lvert\mathcal{T}_{ir}\rvert$	Base	Refinement level	$\lvert\mathcal{T}_{sr}\rvert$	Remeshing error (%)
HORSE	48485	96966	220	6	901120	0.0036
RABBIT	67039	134073	76	6	311296	0.0042
VENUS	50002	100000	80	6	327680	0.0058
FELINE	49919	99732	504	4	129024	0.0131
SKULL	20002	40000	4	8	262144	0.0157

Table 2
Mean differences between $\bar{d}(\mathcal{M}_{sr}, \widehat{\mathcal{M}}_{sr})$ and $\bar{d}(\widehat{\mathcal{M}}_{sr}, \mathcal{M}_{sr})$ according to the bitrate per irregular vertex (bits/iv), computed on 5 typical models (HORSE, RABBIT, VENUS, SKULL and FELINE)

Bitrate (bits/iv)	< 1	1–2	2–6	6–10	> 10
$\bar{d}(\mathcal{M}_{sr}, \widehat{\mathcal{M}}_{sr})$	1.07e–1	1.30e–2	5.34e–3	1.72e–3	1.43e–3
$\bar{d}(\widehat{\mathcal{M}}_{sr}, \mathcal{M}_{sr})$	1.02e–1	1.28e–2	5.34e–3	1.72e–3	1.43e–3
Difference (%)	3.680	1.890	0.291	0.261	0.074

4.4. Second assumption: densely sampled meshes

Let us study the difference of "symmetry" between the distances $\bar{d}(\mathcal{M}_{sr}, \widehat{\mathcal{M}}_{sr})$ and $\bar{d}(\widehat{\mathcal{M}}_{sr}, \mathcal{M}_{sr})$. Table 2 presents a mean of the relative errors between these two distances, computed on 5 typical models (HORSE, RABBIT, VENUS, SKULL and FELINE), and according to different bitrate ranges. The difference being very low ($< 4\%$) for each bitrate range, we can assume that $\bar{d}(\mathcal{M}_{sr}, \widehat{\mathcal{M}}_{sr}) \simeq \bar{d}(\widehat{\mathcal{M}}_{sr}, \mathcal{M}_{sr})$, and we can simplify the computation of D_T by using only one of the unilateral distances:

$$D_T \simeq \bar{d}(\widehat{\mathcal{M}}_{sr}, \mathcal{M}_{sr})^2, \tag{6}$$

or equivalently,

$$D_T \simeq \frac{1}{|\widehat{\mathcal{M}}_{sr}|} \iint_{p \in \widehat{\mathcal{M}}_{sr}} d(p, \mathcal{M}_{sr})^2 \, \mathrm{d}\widehat{\mathcal{M}}_{sr}. \tag{7}$$

D_T should be computed analytically in each point $p \in \widehat{\mathcal{M}}_{sr}$. However, since a *normal mesh* is densely sampled, the number of vertices is large. Thus, we can assume a uniform distribution of the vertices on the surface. Consequently, the integral in (7) can be numerically approximated by a discrete sum (Gersho, 1979). Moreover, the area of triangles being very small relative to global surface, the distance point-surface $d(p, \mathcal{M}_{sr})$ can be computed only from the vertices (Aspert et al., 2002):

$$D_T \simeq \frac{1}{|\widehat{\mathcal{V}}_{sr}|} \sum_{\hat{v} \in \widehat{\mathcal{V}}_{sr}} d(\hat{v}, \mathcal{M}_{sr})^2, \tag{8}$$

with $\hat{v} = Q(v)$ the quantized version of the vertex v, and $|\widehat{\mathcal{V}}_{sr}|$ the number of vertices of $\widehat{\mathcal{M}}_{sr}$ (or equivalently $\mathcal{M}_{sr}$).

Now, we have to deal with $d(\hat{v}, \mathcal{M}_{sr}) = \|\hat{v} - \mathrm{Proj}_{\mathcal{M}_{sr}}(\hat{v})\|_2$.

4.5. Third assumption: an optimal bitrate coding

Let us introduce the quantization error vector $\varepsilon(v) = v - \hat{v}$, between a vertex v and its quantized version $\hat{v}$. Under the assumption of an optimal bitrate coding and in the considered bitrate range, we can assume that the quantization of the coarser levels does not modify consequently the computation of the local coordinate systems in which the details of finer levels are expressed. This introduces further tangential components, but these components remain small compared to normal components. As a result, and since we use a normal remesher, most of error vectors $\varepsilon(v)$ lie in the normal direction at the surface $\mathcal{M}_{sr}$ in v. Therefore, during the computation of $d(\hat{v}, \mathcal{M}_{sr}) = \|\hat{v} - \text{Proj}_{\mathcal{M}_{sr}}(\hat{v})\|_2$, the orthogonal projection of $\hat{v}$ over $\mathcal{M}_{sr}$ remains very close to v:

$$\text{Proj}_{\mathcal{M}_{sr}}(\hat{v}) \simeq v.$$

Finally, we can state that

$$d(\hat{v}, \mathcal{M}_{sr}) = \left\|\hat{v} - \text{Proj}_{\mathcal{M}_{sr}}(\hat{v})\right\|_2 \simeq \|\hat{v} - v\|_2 = \left\|Q(v) - v\right\|_2. \tag{9}$$

Using Eq. (9), Eq. (8) can be written as

$$D_T \simeq \frac{1}{|\mathcal{V}_{sr}|} \sum_{v \in \mathcal{V}_{sr}} \left\|Q(v) - v\right\|_2^2. \tag{10}$$

We notice that the right-hand side of (10) corresponds to the quantization error of the normal mesh geometry, i.e., the MSE denoted by σ^2_{Qsr}. Thus, in case of densely sampled meshes and under the assumption of an optimal bitrate coding, the MSE of the geometry quantization should be a suitable distortion criterion to evaluate the reconstruction error between the irregular input mesh and the quantized one. Finally, we can write

$$D_T \simeq \sigma^2_{Qsr}. \tag{11}$$

This formulation is computed in the euclidean space, and depends on the vertices of the mesh. Now, the proposed bit allocation is processed on the wavelet coefficient subbands. Thus, we have to express the MSE of quantization of the *normal mesh* geometry according to the quantized coefficients.

5. MSE across a wavelet coder

We have shown in Section 4 that the MSE of normal mesh quantization is a suitable criterion to evaluate the reconstruction error between the irregular input mesh and the quantized normal one. In the proposed coder, the unlifted butterfly wavelet transform is applied on the normal mesh $\mathcal{M}_{sr}$. Hence, we obtain the coarse base mesh $\mathcal{M}_0$, and N three-dimensional wavelet coefficient subbands. The geometry $\mathcal{V}_0$ of the coarse mesh $\mathcal{M}_0$ is called the *low frequency* subband. The sets $\mathcal{D}_i$ defined in Section 2 are now the *high frequency* or wavelet coefficients subbands, with i the resolution level.

In (Usevitch, 1996; Payan and Antonini, 2005), it is shown that the MSE of a multidimensional signal encoded across a wavelet coder using a N-level decomposition is equivalent to a weighted sum of the

MSE σ^2_{Qi} introduced by the quantization of each wavelet coefficient subband i. Therefore, the MSE σ^2_{Qsr} between a normal mesh and its quantized version can be written as

$$\sigma^2_{Qsr} = \sum_{i=0}^{N} w_i \sigma^2_{Qi}, \tag{12}$$

where σ^2_{Qi} is the MSE due to the quantization of the wavelet coefficient subband i, and $\{w_i\}$ are the weights due to the biorthogonality of the wavelet transform. The weights relative to the unlifted butterfly wavelet transform are computed in (Payan and Antonini, 2005). They are given by

$$w_i = \frac{|\mathcal{D}_i|}{|\mathcal{V}_{sr}|}\left(\frac{169}{256}\right)^{N-i}, \tag{13}$$

where $|\mathcal{D}_i|$ is the number of coefficients of the subband $\mathcal{D}_i$, and $|\mathcal{V}_{sr}|$ the number of semi-regular vertices.

Recall that in our framework each subband of high frequency wavelet coefficients is splitted in two scalar sets, the *tangential* and *normal sets* (see Section 2). Consequently, the MSE σ^2_{Qi} of the ith high frequency subband ($\forall i \neq 0$) is the sum of the MSE $\sigma^2_{Qi,1}$ and $\sigma^2_{Qi,2}$ due to the quantization of the tangential and normal sets:

$$\sigma^2_{Qi} = \sum_{j \in J_i} \sigma^2_{Qi,j} \quad \forall i \neq 0, \tag{14}$$

where J_i is a set of indices defined by $J_i = \{1, 2\}, \forall i \neq 0$.

On the other hand, the low frequency subband does not present specific properties, since it represents a coarse version of the input mesh. Therefore, the low frequency subband will be splitted in three scalar sets, and the MSE σ^2_{Q0} of the low frequency subband is the sum of the three MSE $\sigma^2_{Q0,j}$ due to the quantization on each coordinate set:

$$\sigma_{Q0} = \sum_{j \in J_0} \sigma^2_{Q0,j}, \tag{15}$$

where J_0 is a set of coordinate indices defined by $J_0 = \{1, 2, 3\}$.

Finally, by merging (14) and (15) in (12), the MSE σ^2_{Qsr} relative to the geometry of a semi-regular mesh encoded with a wavelet coder is given by

$$\sigma^2_{Qsr} = \sum_{i=0}^{N} w_i \sum_{j \in J_i} \sigma^2_{Qi,j} \tag{16}$$

with w_i given by (13). The formulation (16) is finally used as distortion measure during the bit allocation process to evaluate the distortion introduced on the reconstructed mesh by the geometry quantization.

6. Optimal bit allocation

6.1. General purpose

The general purpose of the proposed bit allocation is to optimize the trade-off between the global bitrate and the quality of the reconstructed mesh by controlling and minimizing the losses due to the

geometry quantization at a given bitrate. Compared to the algorithm proposed in (Lavu et al., 2003) that optimizes *locally* the trade-off between the bitrate and the quantization error of the coefficients, the proposed bit allocation process aims to *determine the best set of quantization steps* $\{q_{i,j}\}$ *used to quantize the subbands, that minimizes the global reconstruction error* D_T *of the decoded mesh at a given target bitrate* R_{target}. The quantity R_{target} corresponds to the aimed bitrate for the compressed mesh, expressed here in bits per semi-regular vertex. It can be fixed by either the user, or automatically by the computer, depending on the applications or the bandwidth limitations. The principle is the following. The wanted bitrate is given, and then the reconstruction error is minimized for this specific bitrate. Once the allocation processed and the quantization steps computed for this bitrate, the encoding is performed.

The optimization problem can be formulated as follows:

$$(\mathcal{P}) \quad \begin{cases} \text{minimize} & D_T(\{q_{i,j}\}) \\ \text{with constraint} & R_T(\{q_{i,j}\}) = R_{\text{target}}, \end{cases} \tag{17}$$

where R_T is the given total bitrate. By using a lagrangian operator, this constrained allocation problem can be defined by a lagrangian criterion:

$$J_\lambda\big(\{q_{i,j}\}\big) = D_T\big(\{q_{i,j}\}\big) + \lambda\big(R_T\big(\{q_{i,j}\}\big) - R_{\text{target}}\big), \tag{18}$$

with λ the lagrangian operator. By merging the distortion measure (16) proposed in Section 5 with (18), the lagrangian criterion can be developed in:

$$J_\lambda\big(\{q_{i,j}\}\big) = \sum_{i=0}^{N} w_i \sum_{j\in J_i} \sigma^2_{Qi,j}(q_{i,j}) + \lambda\left(\sum_{i=0}^{N}\sum_{j\in J_i} a_{i,j} R_{i,j}(q_{i,j}) - R_{\text{target}}\right), \tag{19}$$

where $\sigma^2_{Qi,j}(q_{i,j})$ and $R_{i,j}$ are respectively the MSE and the bitrate relative to the (i,j)th component set. The coefficients $a_{i,j}$ depend on the subsampling and correspond to the ratio between the size of the (i,j)th component set and the total number of components $(3 \times |\mathcal{V}_{sr}|)$.

6.2. Optimal solution

The solution of this constrained allocation problem can be obtained by differentiating Eq. (19) with respect to the quantization steps $\{q_{i,j}\}$ and λ (first order conditions), or equivalently by solving the following system:

$$\begin{cases} \frac{\partial J_\lambda(\{q_{i,j}\})}{\partial q_{i,j}} = 0, \\ \frac{\partial J_\lambda(\{q_{i,j}\})}{\partial \lambda} = 0. \end{cases} \tag{20}$$

This system can be developed in

$$w_i \frac{\partial \sigma^2_{Qi,j}(q_{i,j})}{\partial q_{i,j}} + \lambda a_{i,j} \frac{\partial R_{i,j}(q_{i,j})}{\partial q_{i,j}} = 0, \tag{21a}$$

$$\sum_{i=0}^{N}\sum_{j\in J_i} a_{i,j} R_{i,j}(q_{i,j}) = R_{\text{target}}. \tag{21b}$$

Finally, we have to solve the following system of $(2N+4)$ equations with $(2N+4)$ unknowns (the set $\{q_{i,j}\}$ and λ):

$$\frac{\frac{\partial \sigma^2_{Q_{i,j}}(q_{i,j})}{\partial q_{i,j}}}{\frac{\partial R_{i,j}(q_{i,j})}{\partial q_{i,j}}} = -\lambda \frac{a_{i,j}}{w_i}, \tag{22a}$$

$$\sum_{i=0}^{N} \sum_{j \in J_i} a_{i,j} R_{i,j}(q_{i,j}) = R_{\text{target}}. \tag{22b}$$

In order to obtain the optimal quantization steps analytically, (22a) requires to be inverted. Unfortunately, this stage is impossible due to the complexity of the equations. To overcome this problem, an iterative algorithm depending on λ is generally proposed.

6.3. Overall algorithm

The optimal solutions of system (22) for the given bitrate R_{target} are then computed thanks to the following overall algorithm:

(1) λ is given. For each set (i, j), compute $q_{i,j}$ that verifies (22a);
(2) while (22b) is not verified, calculate a new λ by dichotomy and return to step 1;
(3) stop.

The computation of the quantization steps $\{q_{i,j}\}$ as solutions of (22a) can be done according to different methods. In the following Section 7, we propose to process this algorithm with an efficient approach thanks to theoretical models for the bitrate and the MSE (Parisot et al., 2003).

7. Model-based approach

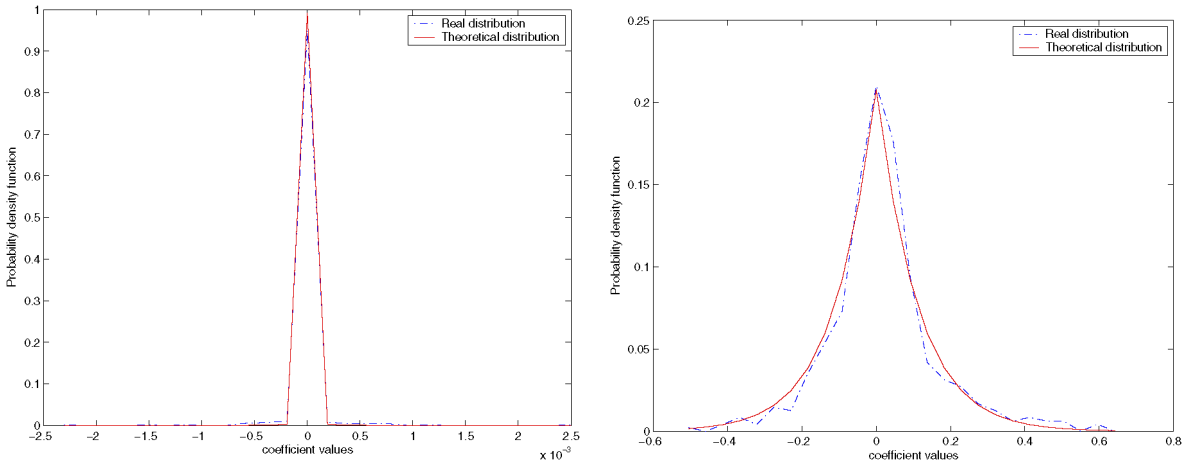
The only way to compute the bitrate and the MSE of the different component sets of the wavelet subbands without real pre-quantizations is to perform a model-based bit allocation. Therefore, we introduce theoretical models for the distortion and the bitrate, depending on the probability density functions of each data set. Let us focus now on the estimation of these density functions.

7.1. Wavelet coefficients distribution

Fig. 5 shows typical probability density functions of the tangential and normal sets of wavelet coefficients of normal meshes obtained by the unlifted butterfly wavelet transform.

We observe that distributions are zero mean and all informations are concentrated on few coefficients (very small variances). By using a χ^2-test, we observe each probability density function of the tangential and normal sets can be modeled by a *Generalized Gaussian Distribution* (GGD).[2]

[2] For instance, the χ^2-test applied on the two probability density functions shown in Fig. 5 provides $\chi^2 = 1.5621$ for the tangential set, and $\chi^2 = 0.1286$ for the normal set when 26 quantization cells are used, meaning that than 99% of the coefficients are well modeled by a GGD.



(a) (b)

Fig. 5. Typical probability density functions of tangential and normal sets (model *Venus*). The dash-dot lines represent the real density functions, and the solid lines represent the corresponding GGD. (a) Tangential set (level 2). (b) Normal set (level 2).

The formulation of a GGD is given by

$$p_{\sigma,\alpha}(x) = a\mathrm{e}^{-|bx|^{\alpha}} \tag{23}$$

with $b = \frac{1}{\sigma}\sqrt{\Gamma(3/\alpha)/\Gamma(1/\alpha)}$ and $a = \frac{b\alpha}{2\Gamma(1/\alpha)}$. The parameter α is computed using the variance σ^2 and the fourth-order moment of each set (Kasner et al., 1999). Fig. 5 also shows the GGD used to model the real distribution (solid lines).

7.2. Processing of the low frequency subband

On the other hand, the three subsets of the low frequency subband do not have any particular distribution and cannot be modeled by an unimodal function like the high frequency component sets, since they represent a coarse version of the original mesh. To overcome this problem, we choose to model and encode the differences between two low frequency components, instead of the components themselves (differential coding) (Payan and Antonini, 2002). We observe that these differences can also be modeled by a GGD. However, this method is interesting if no side information is required by the decoder to reconstruct the good connectivity. This is possible if the differential coding is processed by following the ordered list of low frequency vertices given by the topological coder.

7.3. Theoretical models for the distortion and the bitrate

The theoretical model to compute the MSE σ_Q^2 of a uniform scalar quantizer is given by:

$$\sigma_Q^2 = \int_{-\frac{q}{2}}^{+\frac{q}{2}} p_{\sigma,\alpha}(x)\,\mathrm{d}x + 2\sum_{m=1}^{+\infty} \int_{\frac{q}{2}+|m|q}^{\frac{q}{2}+(|m|-1)q} (x-\hat{x})^2 p_{\sigma,\alpha}(x)\,\mathrm{d}x, \tag{24}$$

where x is an original sample and $\hat{x}$ its decoding value. Furthermore, since an entropy coder is used after the quantization, we suppose that the bitrate R after encoding is equal to the entropy of the quantized components of each set:

$$R = -\sum_{m=-\infty}^{+\infty} \Pr(m) \log_2 \Pr(m). \tag{25}$$

$\Pr(m)$ is the probability of a quantization level m:

$$\Pr(m) = \int_{\frac{q}{2}+|m|q}^{\frac{q}{2}+(|m|-1)q} p_{\sigma,\alpha}(x)\,\mathrm{d}x \quad \text{and} \quad \Pr(0) = \int_{-\frac{q}{2}}^{+\frac{q}{2}} p_{\sigma,\alpha}(x)\,\mathrm{d}x, \tag{26}$$

where $p_{\sigma,\alpha}(x)$ is the probability density function of a subset.

Moreover, the authors of (Parisot et al., 2003) show that for an uniform scalar quantization using the center of the cells as decoding value, the MSE (24) for a GGD $p_{\sigma,\alpha}(x)$ can be rewritten as

$$\sigma_Q^2 = \sigma^2 D(\tilde{q}, \alpha), \tag{27}$$

with σ^2 the variance of the set, and $\tilde{q} = \frac{q}{\sigma}$. $D(\tilde{q}, \alpha)$ is a simple function given by

$$D(\tilde{q}, \alpha) = 1 + 2\sum_{m=1}^{+\infty} (m\tilde{q})^2 f_{0,m}(\tilde{q}, \alpha) - 4\sum_{m=1}^{+\infty} m\tilde{q}\, f_{1,m}(\tilde{q}, \alpha), \tag{28}$$

where functions $f_{n,m}$ are defined by

$$f_{n,m}(\tilde{q}, \alpha) = \int_{\frac{1}{2}\tilde{q}+(m-1)\tilde{q}}^{\frac{1}{2}\tilde{q}+m\tilde{q}} x^n p_{1,\alpha}(x)\,\mathrm{d}x, \tag{29}$$

and by

$$f_{n,0}(\tilde{q}, \alpha) = \int_{-\frac{1}{2}\tilde{q}}^{\frac{1}{2}\tilde{q}} x^n p_{1,\alpha}(x)\,\mathrm{d}x. \tag{30}$$

By the same way, the bitrate R associated to a GGD can be rewritten as (Parisot et al., 2003)

$$R(\tilde{q}, \alpha) = -f_{0,0}(\tilde{q}, \alpha) \log_2 f_{0,0}(\tilde{q}, \alpha) - 2\sum_{m=1}^{+\infty} f_{0,m}(\tilde{q}, \alpha) \log_2 f_{0,m}(\tilde{q}, \alpha). \tag{31}$$

According to the theoretical model (28) for the MSE, and the theoretical model (31) for the bitrate of a component set, the system (22a) becomes

$$h_\alpha(\tilde{q}_{i,j}) = \frac{\frac{\partial D(\tilde{q}_{i,j},\alpha)}{\partial \tilde{q}_{i,j}}}{\frac{\partial R_{i,j}(\tilde{q}_{i,j},\alpha)}{\partial \tilde{q}_{i,j}}} = -\lambda \frac{a_{i,j}}{w_i \sigma_{i,j}^2}, \tag{32a}$$

$$\sum_{i=0}^{N} \sum_{j \in J_i} a_{i,j} R_{i,j}(\tilde{q}_{i,j}, \alpha) = R_{\text{target}}, \tag{32b}$$

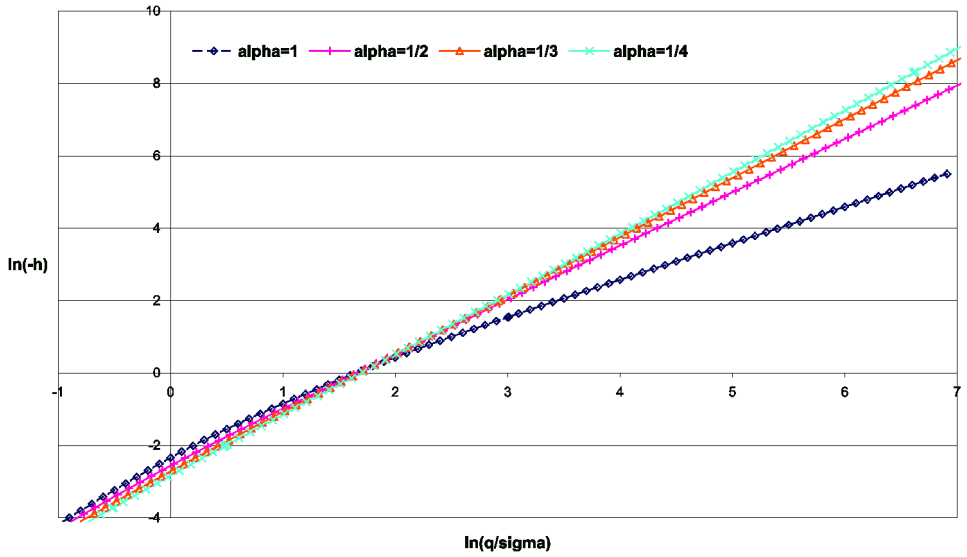


Fig. 6. First LUT used to solve the system (32a): $\ln(-h_\alpha)$ according to $\ln(\tilde{q})$, for different α.

where $h_\alpha(\tilde{q}_{i,j})$ can be developed in

$$h_\alpha(\tilde{q}_{i,j}) = \frac{\sum_{m=1}^{+\infty} m\left[2f_{1,m}(\alpha,\tilde{q}_{i,j}) - 2m\tilde{q}_{i,j}f_{0,m}(\alpha,\tilde{q}_{i,j}) - m\tilde{q}_{i,j}^2\frac{df_{0,m}}{d\tilde{q}_{i,j}}(\alpha,\tilde{q}_{i,j}) + 2\tilde{q}_{i,j}\frac{df_{1,m}}{d\tilde{q}_{i,j}}(\alpha,\tilde{q}_{i,j})\right]}{\frac{p_{1,\alpha}(\tilde{q}_{i,j}/2)}{2}[\ln f_{0,0}(\alpha,\tilde{q}_{i,j})+1] + \sum_{m=1}^{+\infty}\frac{df_{0,m}}{d\tilde{q}_{i,j}}(\alpha,\tilde{q}_{i,j})[\ln f_{0,m}(\alpha,\tilde{q}_{i,j})+1]}\ln 2. \tag{33}$$

7.4. Model-based algorithm

In order to speed the allocation process up, Parisot et al. (2003) propose to use some offline computed Look-Up Tables (LUT) to solve the system (32). They propose to exploit two parametric curves:

- $[\ln(\tilde{q});\ \ln(-h_\alpha)]$: this LUT allows to compute the quantization step q corresponding to a specific $h_\alpha(\tilde{q}_{i,j})$. Fig. 6 shows the parametric curves corresponding to this LUT. This allows to compute the quantization steps verifying (32a).
- $[R;\ln(-h_\alpha)]$: this LUT allows to compute the bitrate R corresponding to a specific $h_\alpha(\tilde{q}_{i,j})$. Fig. 7 shows the parametric curves corresponding to this LUT. This permits to verify the constraint on the bitrate (32b);

In that case, the algorithm given in Section 6.3 becomes:

(1) compute the variance $\sigma_{i,j}^2$ and the parameter $\alpha_{i,j}$ for each set (i, j);
(2) a value of λ is given. For each set (i, j), compute $h_\alpha(\tilde{q}_{i,j})$ thanks to the right-hand side of (32a). Then, use the LUT of $[R;\ln(-h_\alpha)]$ to compute the corresponding bitrate $R_{i,j}$;

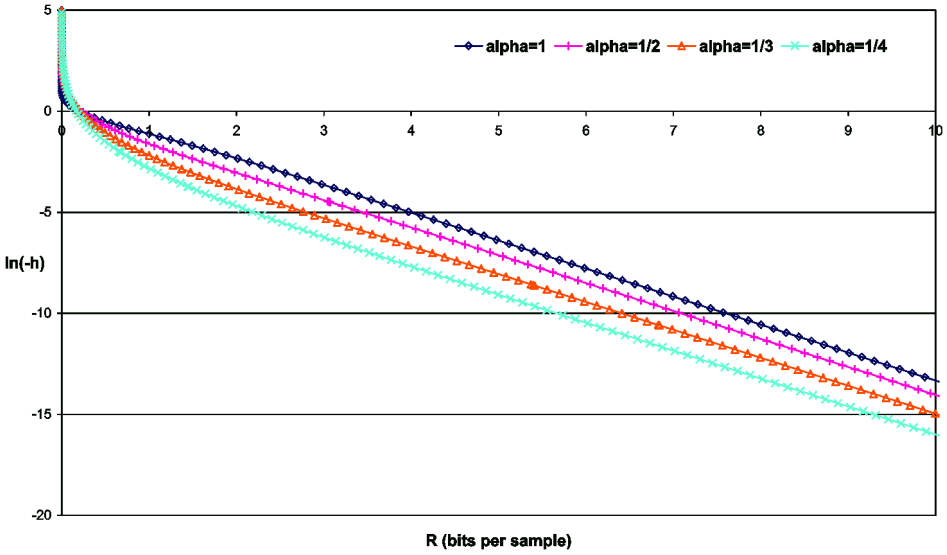


Fig. 7. Second LUT used to solve the system (32a): $\ln(-h_\alpha)$ according to R, for different α.

(3) while (32b) is not verified, calculate a new λ by dichotomy and return to step 2;
(4) At this step, the optimal λ is known. Thus, for each set (i, j), use the LUT of $[\ln(\tilde{q});\ \ln(-h_\alpha)]$ to compute the optimal quantization step $q_{i,j}$ corresponding to the value of $h_\alpha(\tilde{q}_{i,j})$ found in step 2.
(5) stop.

7.5. Complexity

In this section, we evaluate the complexity of the model-based algorithm to show the interest of the proposed approach.

Step 1 of the algorithm permits the computation of the variance σ^2 and of the parameter α. The parameter α is computed from the variance and the fourth-order moment for each component set (Kasner et al., 1999). This step can be done in 4 operations per component.

At step 2, after the computation of $\ln(-h_\alpha)$ using λ, σ^2 and Eq. (32a), the set of $\{R_{i,j}\}$ is computed at low cost by addressing the LUT associated to $[R;\ \ln(-h_\alpha)]$.

Step 3 consists in computing a simple weighted sum of the bitrates estimated at step 2 (2 arithmetic operations per component set) to verify the constraint on the global bitrate. The computation of a new λ is done by a simple dichotomy.

At step 4, the set of quantization steps $\{q_{i,j}\}$ is computed at low cost by addressing the LUT associated to $[\ln(\tilde{q});\ \ln(-h_\alpha)]$.

The convergence of the algorithm is reached after few iterations (lower than 5). Finally, the step 1 represents the highest computational cost of this algorithm, with 4 operations per sample, hence a computational complexity of approximately 12 operations per semi-regular vertex. This involves a fast allocation process with a very low computational complexity, taking less than 0.4 second on a Pentium III 1 GHz, 512 Mbytes RAM.

8. Experimental results

This section presents some experimental results of the proposed coder, and we compare its performances to some state-of-the-art coders. These coders are:

- The zerotree coder of normal meshes (NMC) (Khodakovsky and Guskov, 2002);
- The EQ mesh coder (EQMC) for normal meshes (Lavu et al., 2003);
- The original Zerotree Coder (PGC) (Khodakovsky et al., 2000) for semi-regular meshes, including the remeshing technique MAPS (Lee et al., 1998);

The coders NMC and EQMC are currently the most efficient geometry coders. In order to encode the connectivity of the base mesh, we use the topology coder of Touma and Gotsman (Touma and Gotsman, 1998) as in the three state-of-the-art coders previously denoted. This permits to compare only the performances of the different geometry coders.

Recall that the main objective of our algorithm (but also of EQMC) is to improve the coding performances by optimizing the rate-distortion trade-off, *for one specific target bitrate*. Thus, to confirm that the proposed algorithm achieves performance gains compared to the state-of-the-art coders for any given bitrate, Figs. 8, 9, and 10 show the PSNR curves according to different given bitrates (per irregular vertex), for the models HORSE, RABBIT, and VENUS. The curves are constructed as follows. For our algorithm, they depend on several values of R_{target}, each dot corresponding to a mesh coded and decoded at its finest resolution. For EQMC, they depend on several values of λ (Lavu et al., 2003). For NMC and PGC, they depend on several given bitstream sizes (Khodakovsky and Guskov, 2002; Khodakovsky et al., 2000).

The PSNR is given by

$$\text{PSNR} = 20\log_{10}(\text{bb}/d_s),$$

where bb is the bounding box diagonal and d_s is the surface-to-surface distance between the irregular input mesh and the reconstructed semi-regular one (computed with MESH (Aspert et al., 2002)).

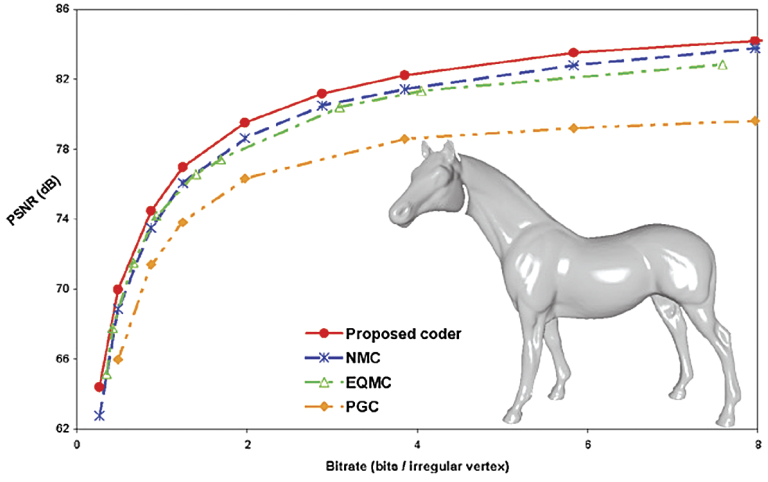


Fig. 8. Bitrate-PSNR curve for HORSE at its finest resolution (901120 triangles).

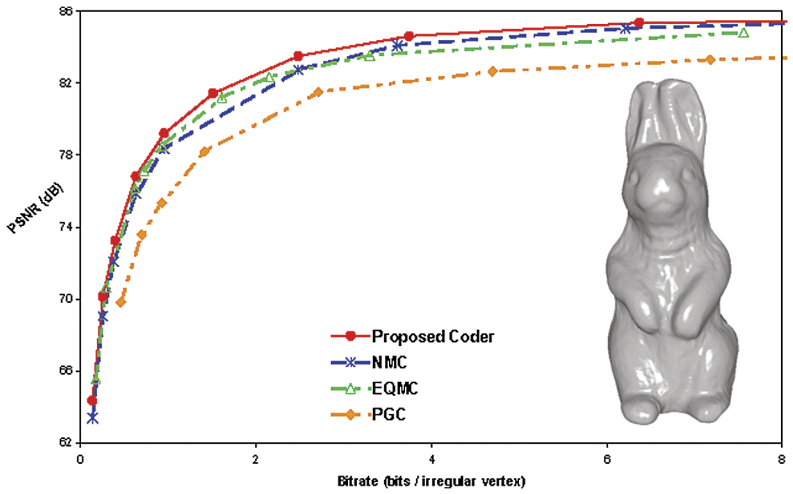


Fig. 9. Bitrate-PSNR curve for RABBIT at its finest resolution (311296 triangles).

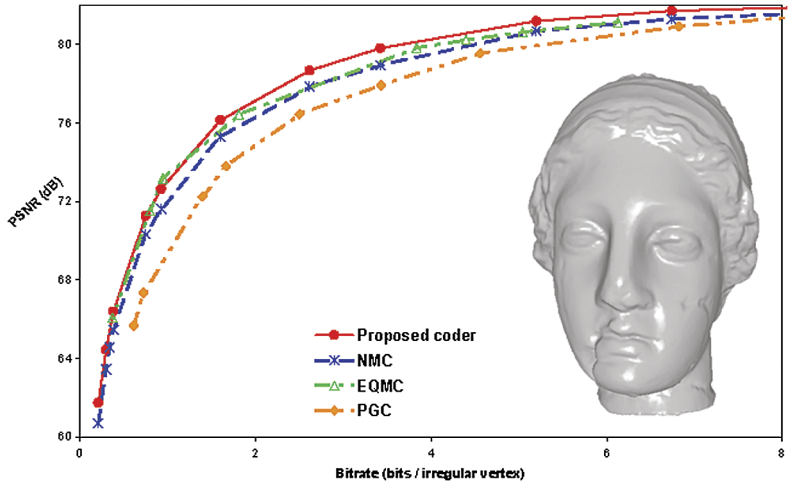


Fig. 10. Bitrate-PSNR curve for VENUS at its finest resolution (327680 triangles).

We observe that, for any target bitrate, the proposed coder provides performance gains compared the state-of-the-art coders NMC and EQMC (excepted one bitrate for the object VENUS where EQMC is slightly better). We obtain similar results with the models FELINE, SKULL and MOLECULE. This is remarkable since theoretically the MSE should be a suitable criterion only in case of optimal rate coding (high bitrates). Finally, we find experimentally that the MSE is always a suitable criterion, for any bitrate range.

Table 3 gives the PSNR values relative to the proposed coder and to the coder NMC according to different given bitrates for all the models (at their finest resolution). We observe the proposed bit allocation improves the coding performances up to +2.5 dB. In addition, Fig. 11 provides some visual benefits relative to the use of the proposed coder. This figure shows the distribution of the reconstruction error on the object FELINE, quantized with the proposed coder (Fig. 11(a)) and with NMC (Fig. 11(b)). The colour

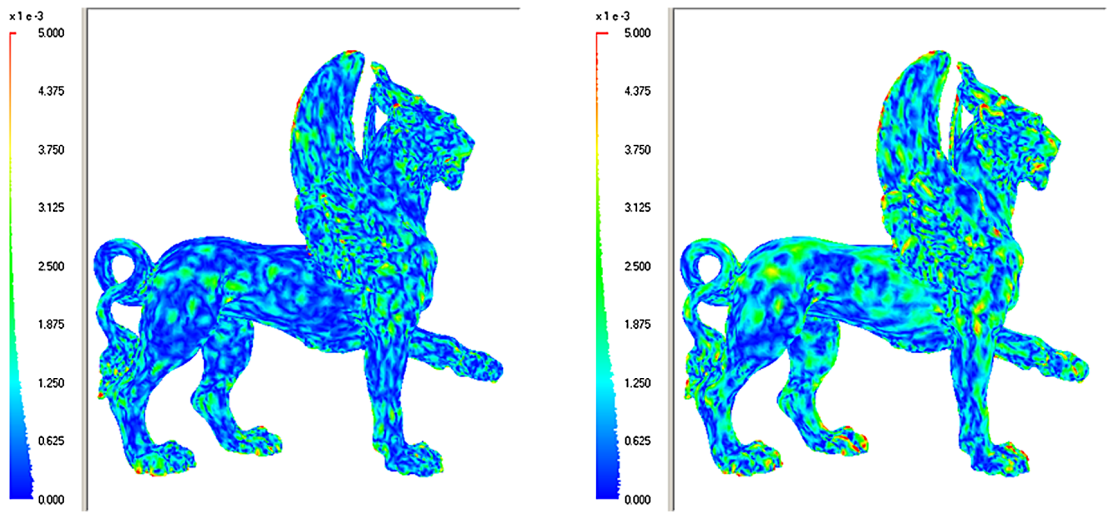
Table 3
Proposed coder versus the state-of-the-art NMC: PSNR improvement (in dB) for 6 typical models, at their finest resolution

RABBIT (155650 vertices, 311296 triangles)								
Bitrate (bits/iv)	0.14	0.26	0.38	0.92	2.48	3.61	6.20	9.03
Proposed coder	64.35	70.15	73.26	79.20	83.50	84.61	85.35	85.50
NMC	63.38	69.04	72.09	77.99	82.74	84.08	85.05	85.42
Improvement	0.97	1.11	1.17	1.21	0.76	0.53	0.30	0.07
HORSE (450562 vertices, 901120 triangles)								
Bitrate (bits/iv)	0.26	0.87	1.24	1.97	2.88	3.85	5.84	11.51
Proposed coder	64.39	74.47	76.97	79.51	81.18	82.22	83.52	84.79
NMC	62.75	73.51	76.06	78.63	80.50	81.43	82.80	84.36
Improvement	1.64	0.96	0.91	0.88	0.68	0.79	0.72	0.43
VENUS (163842 vertices, 327680 triangles)								
Bitrate (bits/iv)	0.20	0.34	0.92	1.60	3.42	5.20	6.74	9.06
Proposed coder	61.75	65.64	72.56	76.14	79.82	81.21	81.73	82.02
NMC	60.71	64.59	71.63	75.29	78.95	81.31	81.77	80.69
Improvement	1.03	1.05	1.02	0.86	0.87	0.52	0.41	0.25
FELINE (64510 vertices, 129024 triangles)								
Bitrate (bits/iv)	0.39	0.71	1.00	1.30	2.22	3.28	4.37	5.05
Proposed coder	59.34	65.35	68.10	70.60	73.72	74.84	75.14	75.22
NMC	57.03	62.86	66.57	68.99	72.99	74.47	74.96	75.11
Improvement	2.32	2.49	1.53	1.62	0.75	0.36	0.18	0.11
SKULL (131074 vertices, 262144 triangles)								
Bitrate (bits/iv)	0.21	0.53	0.82	1.52	2.25	4.08	5.08	7.90
Proposed coder	54.60	61.19	64.70	68.38	70.65	72.96	73.38	73.58
NMC	53.75	60.98	63.96	67.73	70.27	72.60	73.19	73.54
Improvement	0.85	0.21	0.74	0.66	0.39	0.36	0.19	0.04
MOLECULE (54272 vertices, 108544 triangles)								
Bitrate (bits/iv)	0.43	0.89	1.69	2.25	3.10	6.08	7.60	10.79
Proposed coder	44.69	52.00	57.55	59.53	61.54	64.30	65.38	66.15
NMC	43.57	50.93	56.30	58.48	60.73	63.43	64.59	65.59
Improvement	1.12	1.07	1.24	1.05	0.81	0.87	0.79	0.55

corresponds to the magnitude of the distance point-surface normalized by the bounding box diagonal, between the input irregular mesh and the quantized one (computed with MESH (Aspert et al., 2002)) (for colours see the web version of this article). One can argue that NMC leads to more local errors than the proposed algorithm. Moreover, Fig. 12 shows renderings of VENUS, compressed at different given bitrates. This demonstrates that even at low bitrates the meshes quantized with the proposed algorithm is not so far from the original irregular one.

9. Conclusions

In this paper, we design an original wavelet coder based on the rate-distortion optimization, for densely sampled triangular meshes. Exploiting the *normal remesher* of Guskov et al. (2000), our coder includes an original model-based bit allocation that optimizes the quantization of the wavelet coefficients at any



(a) (b)

Fig. 11. Distribution of the reconstruction error on the object FELINE at its finest resolution. The total bitrate is equal to 0.71 bits/irregular vertex. (a) Proposed method. PSNR = 65.35 dB. (b) NMC. PSNR = 62.86 dB.

given bitrate. By assuming that the quantization of the coarser levels does not modify significantly the computation of the local coordinate systems, we argue that the weighted sum of the MSE relative to the quantization of each wavelet component set is a suitable distortion criterion to evaluate the reconstruction error between the irregular input mesh and the reconstructed normal one during the bit allocation. By minimizing this MSE for a given target bitrate, the allocation process dispatches the bit budget across the wavelet subbands according to their influence on the quality of the reconstructed mesh for this specific bitrate. Moreover, the use of theoretical models for the distortion and the bitrate of each component set involves a very fast computation of the optimal quantization steps. Experimental results demonstrate that, for any given bitrate, the proposed approach provides improvements in coding performance compared to the two state-of-the-art normal mesh coders (Khodakovsky and Guskov, 2002; Lavu et al., 2003) (up to $+2.5$ dB compared to the original zerotree coder), for a very low computational complexity. In future works, we could improve this algorithm to allow a strictly progressive compression, since our method is at moment only scale-wise progressive.

Acknowledgements

Datasets are courtesy of Cyberware, Headus, The Scripps Research Institute, and University of Washington. We are particularly grateful to Igor Guskov for providing us with his normal meshes and his executable NMC, and Shridar Lavu for providing us with his executable EQMC. We also want to acknowledge the anonymous reviewers for their advises which permitted to improve the quality of the paper.

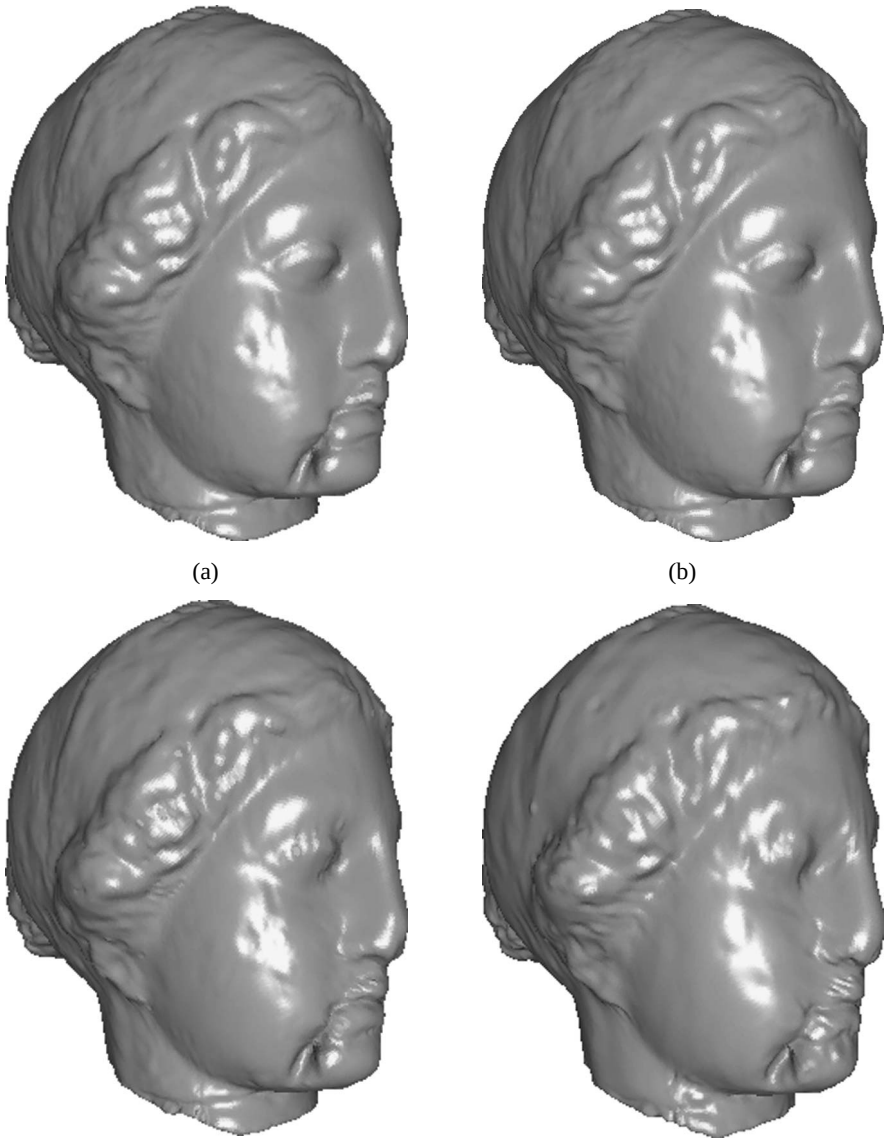


Fig. 12. Renderings of VENUS at its finest resolution, compressed at different target bitrates. (a) Input mesh. (b) 2.6 bits/irregular vertex (bitstream size = 16340 bytes). (c) 0.7 bits/irregular vertex (bitstream size = 4677 bytes). (d) 0.3 bits/irregular vertex (bitstream size = 1842 bytes).

References

Alliez, P., Gotsman, C., 2003. Recent advances in compression of 3D meshes. In: Proceedings of the Symposium on Multiresolution in Geometric Modeling.

Aspert, N., Santa-Cruz, D., Ebrahimi, T., 2002. Mesh: Measuring errors between surfaces using the Hausdorff distance. In: Proceedings of the IEEE ICME, vol. I, pp. 705–708.

Chow, M., 1997. Optimized geometry compression for real-time rendering. In: Proceedings of the 8th Conference on Visualization '97. IEEE Computer Society Press, pp. 347–354.

Cignoni, P., Rocchini, C., Scopigno, R., 1998. Metro: Measuring error on simplified surfaces. Computer Graphics Forum 2 (17), 167–174.

Gersho, A., 1979. Asymptotically optimal block quantization. IEEE Trans. Inform. Theory (25), 373–380.

Gotsman, C., Gumhold, S., Kobbelt, L., 2002. Simplification and compression of 3D meshes. In: Iske, M.F.A., Quak, E. (Eds.), Tutorials on Multiresolution in Geometric Modelling. In: Munich Summer School Lecture Notes.

Gu, X., Gortler, S.J., Hoppe, H., 2002. Geometry images. In: Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques. ACM Press, pp. 355–361.

Guskov, I., Vidimce, K., Sweldens, W., Schröder, P., 2000. In: Computer Graphics Proceedings, pp. 95–102.

Karni, Z., Gotsman, C., 2000. Spectral compression of mesh geometry. In: ACM SIGGRAPH Conference Proceedings, pp. 279–286.

Kasner, J., Marcellin, M., Hunt, B., 1999. Universal trellis coded quantization. IEEE Trans. Image Process. 8 (12), 1677–1687.

Khodakovsky, A., Guskov, I., 2002. Normal Mesh Compression, Geometric Modeling for Scientific Visualization. Springer-Verlag.

Khodakovsky, A., Schröder, P., Sweldens, W., 2000. Progressive geometry compression. In: Proceedings of SIGGRAPH 2000.

King, D., Rossignac, J., 1999. Optimal bit allocation in 3D compression. J. Comput. Geom. 14 (1–3), 99–118.

Lavu, S., Choi, H., Baraniuk, R., 2003. Geometry compression of normal meshes using rate-distortion algorithms. In: Proceedings of the Eurographics/ACM SIGGRAPH Symposium on Geometry Processing.

Lee, A., Moreton, H., Hoppe, H., 2000. Displaced subdivision surfaces. In: Proceedings of ACM SIGGRAPH 2000, pp. 85–94.

Lee, A., Sweldens, W., Schröder, P., Cowsar, P., Dobkin, D., 1998. MAPS: Multiresolution adaptive parameterization of surfaces. In: Proceedings of SIGGRAPH'98.

Levoy, M., 1999. The digital Michelangelo project. In: Proceedings of the 2nd International Conference on 3D Digital Imaging and Modeling.

Li, J., Li, J., Kuo, C.C., 1997. Progressive compression of 3D graphic models. In: International Conference on Multimedia Computing and Systems, pp. 135–142.

Lopresto, S., Ramchandran, K., Orchard, M., 1997. Image coding based on mixture modeling of wavelet coefficients and a fast estimation-quantization framework. In: Proceedings of IEEE Data Compression Conference (DCC), Snowbird, Utah, pp. 221–226.

Luebcke, D., Halle, B., 2001. Perceptually driven simplification for interactive rendering. In: Proceedings of the 12th Eurographics Workshop on Rendering Techniques.

Parisot, C., Antonini, M., Barlaud, M., 2003. 3D scan based wavelet transform and quality control for video coding. EURASIP J. Appl. Signal Process. (Special issue on multimedia Signal Processing).

Payan, F., Antonini, M., 2002. Multiresolution 3D mesh compression. In: Proceedings of IEEE International Conference in Image Processing (ICIP), Rochester, USA.

Payan, F., Antonini, M., 2003a. Weighted bit allocation for multiresolution 3D mesh geometry compression. In: Proceedings of SPIE Visual Communications and Image Processing (VCIP) Conference, Lugano, Switzerland, pp. 1947–1956.

Payan, F., Antonini, M., 2003b. 3D multiresolution context-based coding for geometry compression. In: Proceedings of IEEE International Conference in Image Processing (ICIP), Barcelona, Spain, pp. 785–788.

Payan, F., Antonini, M., 2005. MSE approximation for model-based compresion of multiresolution semiregular meshes. In: Proceedings of IEEE EUSIPCO 2005, 13th European Conference on Signal Processing, Antalya, Turkey, September 2005.

Schröder, P., Sweldens, W., 1995. Spherical wavelets: Efficiently representing functions on the sphere. Proc. SIGGRAPH 95, 161–172.

Sim, J.-Y., Kim, C.-S., Kuo, C.J., Lee, S.-U., 2002. Normal mesh compression based on rate-distortion optimization. In: Proceedings of the International Workshop on MultiMedia Signal Processing.

Sorkine, O., Cohen-Or, D., Toledo, S., 2003. High-pass quantization for mesh encoding. In: Proceedings of the Eurographics/ACM SIGGRAPH Symposium on Geometry Processing. Eurographics Association, pp. 42–51.

Sweldens, W., 1998. The lifting scheme: A construction of second generation wavelets. SIAM J. Math. Anal. 29 (2), 511–546.

Touma, C., Gotsman, C., 1998. Triangle mesh compression. In: Graphics Interface'98, pp. 26–34.

Usevitch, B., 1996. Optimal bit allocation for biorthogonal wavelet coding, in: IEEE Data Compression Conference.

Zorin, D., Schröder, P., Sweldens, W., 1997. Interactive multiresolution mesh editing. Computer Graphics (Annual Conference Series) 31, 259–268.