MSE APPROXIMATION FOR MODEL-BASED COMPRESSION OF MULTIRESOLUTION SEMIREGULAR MESHES

Frédéric Payan and Marc Antonini

Laboratoire I3S - UMR 6070 CNRS - Université de Nice - Sophia Antipolis
Route des Lucioles - F-06903 Sophia Antipolis FRANCE
phone: +33 (0)4 92 94 27 22, fax: +33 (0)4 92 94 28 98, email: {fpayan,am}@i3s.unice.fr
web: www.i3s.unice.fr/~barlaud/Creative.htm

ABSTRACT

It is well known in wavelet image coding that the biorthogonality of wavelet filters influences the estimation of the Mean Square Error (MSE) due to data quantization. It has been indeed shown that the MSE can be approximated by a weighted MSE with the weights depending on the synthesis filters [1, 2]. These works have been carried on canonical sampling grid for 1D signals or 2D images. In this paper we propose an elegant solution to compute the weights in case of 3D meshes sampled on a triangular grid. We show that the weights can depend only on the polyphase components of the synthesis filter bank, allowing an easy computation for filters based on a lifting scheme. In particular, we compute the weights for the well-known Butterfly lifting scheme. Then we show that using the proposed MSE approximation improves the performances of a model-based bit allocation process. We obtain an encoding gain up to 3.5 dB compared to the state-of-the-art zerotree coders.

1. INTRODUCTION

Today semiregular remeshing and wavelet transforms are more and more exploited in mesh processing to perform efficient compression methods [3, 4]. A wavelet-based coder includes generally a bit allocation process dispatching the bits across the wavelet subbands, in order to improve the data compression. One method frequently used is to minimize the MSE of the data quantization for a given target bitrate. In order to speed the allocation process up, the MSE can be approximated by a weighted sum of the MSE of each coefficient subband. It is indeed shown that using biorthogonal filters weights the amount of quantization error which appears on a reconstructed data, and that the weights depend on the coefficients of the synthesis filters [1, 2]. In this paper, we develop the formulation of the weighted MSE for a triangular edge lattice. Moreover, we show that the weights can depend only on the polyphase components of the synthesis filter bank, which is very useful in case of lifting schemes. Experimentally, we show that using this MSE approximation in a model-based bit allocation improves the coding performance of a wavelet coder for any semiregular mesh (obtained with [5] or [6]) and any version of the Butterfly-based lifting scheme (lifted or unlifted). The proposed coder finally outperforms the state-of-the-art zerotree coders [3, 4].

The remainder of this paper is organized as follows. Section 2 develops the MSE approximation of a mesh geometry quantized with a wavelet coder. Section 3 provides the weight values for the Butterfly-based lifting schemes. Then, we experiment the effects of these weights thanks to a model-based bit allocation for triangular mesh coders in section 4, and conclude in section 5.

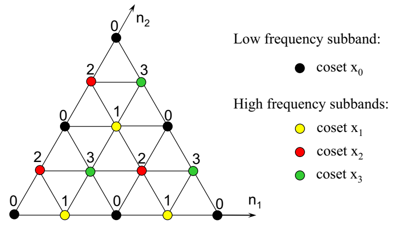

Figure 1: Downsampling grid of the triangular edge lattice.

2. MSE APPROXIMATION ON A TRIANGULAR EDGE LATTICE

2.1 Challenge

A semiregular mesh is based on a triangular edge lattice [7] (see fig. 1). A wavelet transform for meshes corresponds consequently to a 4-channel filter bank. Hence, the geometry of a semiregular mesh M is transformed into 4 cosets $\{x_i, i = 0,...,3\}$ on account of an analysis filter bank $\{h_i, i = 0,...,3\}$ and a downsampling $\downarrow D$ (see fig. 2). The cosets are then quantized and the quantization error ε_i between the i^{th} coset x_i and its quantized value $\hat{x}_i$ is given by:

$$\varepsilon_i = (x_i - \hat{x}_i). \quad (1)$$

This formulation corresponds to the additive noise model of quantizers given by [8]. An upsampling $\uparrow D$ followed by a synthesis wavelet transform g_i provides the geometry of the reconstructed mesh $\hat{M}$. The challenge of this section is to obtain the MSE of the quantized mesh geometry according to the quantization error of each wavelet subband and the knowledge of the synthesis filter bank.

2.2 MSE of a quantized mesh

In order to simplify the derivation, let us consider the source mesh M as a realization of a stationary and ergodic random process [8]. So the total quantization error ε can be

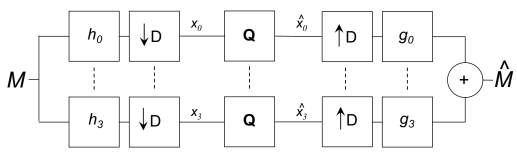

Figure 2: Principle of a 4-channel wavelet coder.

considered as a deterministic quantity, and is defined by $\varepsilon = \{\varepsilon(\mathbf{k}) = (M(\mathbf{k}) - \hat{M}(\mathbf{k})) \in \mathbb{R} \mid \mathbf{k} \in K\}$, K being the sampling grid. K is given by $K = \Gamma\mathbb{Z}^2$ with Γ an inversible matrix permitting to obtain the datas 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 [7]. Therefore the MSE between the geometry of the source mesh and the quantized one can be written as:

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

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 exploiting

$$R_{\mathscr{E}}(\mathbf{z}) = \mathscr{E}(\mathbf{z})\,\mathscr{E}(\mathbf{z}^{-1}), \tag{3}$$

with $\mathscr{E}(\mathbf{z})$ the z-transform of ε, and $\mathbf{z} = (z_1, z_2)$. According to Fig. 2, $\mathscr{E}(\mathbf{z})$ can be formulated in function of the error of each coset s_i [9]:

$$\mathscr{E}(\mathbf{z}) = \sum_{i=0}^{3} G_i(\mathbf{z})\,\mathscr{E}_i(\mathbf{z}^D), \tag{4}$$

with $\mathscr{E}_i(\mathbf{z})$ and $G_i(\mathbf{z})$ respectively the z-transform of ε_i related to s_i, and of the synthesis filter g_i. D is the dilation matrix relative to the upsampling [7], $\mathbf{z}^D = \{\mathbf{z}^{\mathbf{d}_1}, \mathbf{z}^{\mathbf{d}_2}\}$, with $\mathbf{d}_j$ the j^{th} column vector of the matrix D, and $\mathbf{z}^{\mathbf{d}_j}$ given by:

$$\mathbf{z}^{\mathbf{d}_j} = \prod_{n=1}^{2} z_n^{\mathbf{d}_j(n)}. \tag{5}$$

By assuming there is no cross-correlation between errors $\varepsilon_i(\mathbf{k})$ and $\varepsilon_i(\mathbf{k}')$ (for all $\mathbf{k} \neq \mathbf{k}'$) [8], we can write

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

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

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

Hence, Eq. (3) and (4) provide:

$$R_{\mathscr{E}}(\mathbf{z}) = \sum_{i=0}^{3} R_{G_i}(\mathbf{z})\,R_{\mathscr{E}_i}(\mathbf{z}^D). \tag{6}$$

Applying the inverse z-transform on Eq. (6) yields the formulation of the autocorrelation function of the reconstruction error:

$$r_\varepsilon(\mathbf{t}) = \sum_{i=0}^{3}\sum_{\tau} r_{g_i}(\tau)\,r_{\varepsilon_i}(D\mathbf{t} - \tau). \tag{7}$$

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

$$r_\varepsilon(\mathbf{0}) = \sum_{i=0}^{3}\sum_{\tau} r_{g_i}(\tau)\,r_{\varepsilon_i}(-\tau). \tag{8}$$

By assuming that the quantization error samplings are uncorrelated [8], $r_{\varepsilon_i}(-\tau) = 0$ if $\tau \neq \mathbf{0}$, and consequently,

$$r_\varepsilon(\mathbf{0}) = \sum_{i=0}^{3} r_{g_i}(\mathbf{0})\,r_{\varepsilon_i}(\mathbf{0}). \tag{9}$$

Now, the problem is to deal with $r_{g_i}(\mathbf{0})$ and $r_{\varepsilon_i}(\mathbf{0})$.

2.3 Energy of the synthesis filter

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

$$r_{g_i}(\mathbf{0}) = \frac{1}{2\pi j}\oint_\Gamma G_i(\mathbf{z})\,G_i(\mathbf{z}^{-1})\,\mathbf{z}^{-1}\,d\mathbf{z}. \tag{10}$$

According to the downsampling grid (see fig. 1), a synthesis filter bank $\{g_i\}$ on a triangular edge lattice can be formulated according to the polyphase notation:

$$G_i(\mathbf{z}) = \sum_{j=0}^{3} \mathbf{z}^{-\mathbf{t}_j}\,G_{i,j}(\mathbf{z}^D) \quad \text{for } i \in \{0,...,3\}, \tag{11}$$

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

$$G_{i,j}(\mathbf{z}) = \sum_{\mathbf{k}\in\mathbb{Z}^2} g_i\,(D\mathbf{k} + \mathbf{t}_j)\,\mathbf{z}^{-\mathbf{k}}, \tag{12}$$

and $\mathbf{z}^{-\mathbf{t}_j}$ the shift relative to the j^{th} coset [7]. By exploiting (11) and (12), (10) can be developed in:

$$r_{g_i}(\mathbf{0}) = \frac{1}{2\pi j}\sum_{u=0}^{3}\sum_{v=0}^{3}\sum_{\mathbf{k}\in\mathbb{Z}^2}\sum_{\mathbf{k}'\in\mathbb{Z}^2} g_i\,(D\mathbf{k} + \mathbf{t}_u) \\ g_i\,(D\mathbf{k} + \mathbf{t}_v)\oint_\Gamma \mathbf{z}^{(-D\mathbf{k} + D\mathbf{k}' - \mathbf{t}_u + \mathbf{t}_v - \mathbf{1})}\,d\mathbf{z}. \tag{13}$$

Using Cauchy theorem, that is,

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

the integral operator of Eq. (13) is equal to $\mathbf{1}$ if $(-D\mathbf{k} + D\mathbf{k}' - \mathbf{t}_u + \mathbf{t}_v = \mathbf{0})$ is satisfied. The dilation matrix D being invertible, this condition becomes

$$(-\mathbf{k} + \mathbf{k}') - (D^{-1}\mathbf{t}_u - D^{-1}\mathbf{t}_v) = \mathbf{0}. \tag{14}$$

By studying the definition domains of the variables [7], we find that the only set of solutions of the condition (14) 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

$$r_{g_i}(\mathbf{0}) = \sum_{j=0}^{3}\sum_{\mathbf{k}\in\mathbb{Z}^2} g_i\,(D\mathbf{k} + \mathbf{t}_j)^2, \tag{15}$$

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.

2.4 Energy of the quantization error

By assuming that the quantization error samplings are uncorrelated [8], the energy $r_{\varepsilon_i}(\mathbf{0})$ is:

$$r_{\varepsilon_i}(\mathbf{0}) = \sum_{\mathbf{k}\in\mathbb{Z}^2} \varepsilon_i(\mathbf{k})^2 = N_{s_i}\sigma_{\varepsilon_i}^2, \tag{16}$$

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

2.5 Solution for a one-level decomposition

Finally, by merging (15), (16) and (9) in (2), we obtain the expression of the MSE of a reconstructed mesh :

$$\sigma_\varepsilon^2 = \sum_{i=0}^{3} \frac{N_{s_i}}{N_s} w_i \sigma_{\varepsilon_i}^2 \quad \text{with} \quad w_i = \sum_{j=0}^{3} \sum_{\mathbf{k}\in\mathbb{Z}^2} g_{i,j}(\mathbf{k})^2 . \qquad (17)$$

where $g_{i,j}(\mathbf{k})$ represents the coefficient k of the j^{th} polyphase component of the synthesis filter i. Note that the proposed formulation allows to deduce *the weights w_i from only the coefficients $\{g_{i,j}(\mathbf{k})\}$ of the polyphase matrix components of the synthesis filters*. This is useful for a lifting scheme, since the weights can be obtained directly from the polyphase components, without computing the synthesis filter bank.

2.6 Solution for a N-level decomposition

Mostly wavelet coders exploit several levels of decomposition by applying several times the wavelet transform on the coset of lowest frequency. By the same way as for the one-level decomposition, the MSE across a N-level wavelet coder can be approximated by:

$$\boxed{\sigma_\varepsilon^2 = \frac{N_{s_{N-1,0}}}{N_s} W_{N-1,0}\sigma_{\varepsilon_{N-1,0}}^2 + \sum_{i=0}^{N-1}\sum_{l=1}^{3} \frac{N_{s_{i,l}}}{N_s} W_{i,l}\sigma_{\varepsilon_{i,l}}^2 ,} \qquad (18)$$

where $N_{s_{i,j}}$, $\sigma_{\varepsilon_{i,j}}^2$, and $W_{i,l}$ represent respectively the number of samples, the MSE, and the weights relative to the coset (i,j), with i the level of decomposition and j the channel index[1]. The weights $W_{i,l}$ depend on the weights $\{w_i\}$ defined for a one-level decomposition:

$$\boxed{W_{i,l} = (w_0)^i w_l .} \qquad (19)$$

3. COMPUTATION OF WEIGHTS FOR THE BUTTERLFY-BASED LIFTING SCHEME

This lifting scheme exists in two different versions: the classical *lifted* version (a prediction step and an update step), and the *unlifted* version (only a prediction step). The description of this lifting scheme can be found in [10].

3.1 Polyphase matrix for a 4-channel lifting scheme

As we said in the previous section, only the polyphase components are needed to compute the weights. In the case of a 4-channel lifting scheme, the polyphase matrix is [7]:

$$G = \begin{pmatrix} 1 & p_1 & p_2 & p_3 \\ -u_1 & 1-u_1p_1 & -u_1p_2 & -u_1p_3 \\ -u_2 & -u_2p_1 & 1-u_2p_2 & -u_2p_3 \\ -u_3 & -u_3p_1 & -u_3p_2 & 1-u_3p_3 \end{pmatrix} \qquad (20)$$

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 (17) allows to compute directly the corresponding weights w_i, without designing the synthesis filter bank.

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 (20), are:

$$\begin{cases} w_0 & = \frac{169}{256} \simeq 0.66015625 \\ w_1 = w_2 = w_3 & = \frac{1727}{2048} \simeq 0.843261715. \end{cases} \qquad (21)$$

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

$$\begin{cases} w_0 & = \frac{169}{256} \simeq 0.66015625 \\ w_1 = w_2 = w_3 & = 1. \end{cases} \qquad (22)$$

4. EXPERIMENTAL RESULTS

To show the interest of taking into account the biorthogonality of the Butterfly-based filters in a bit allocation process, we experiment the coder presented in [11] (see fig. 3).

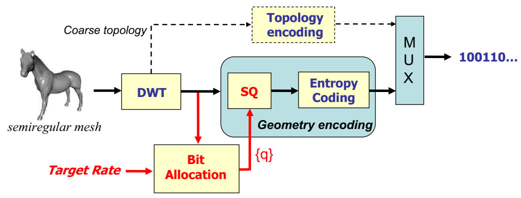

Figure 3: Proposed coder for semiregular meshes.

The source mesh M (obtained thanks to a remeshing method like [5] or [6]) is transformed in several subbands of wavelet coefficients thanks to the Butterfly-based lifting scheme. The coordinates of the wavelet coefficients are then encoded with scalar quantizers *SQ* (depending on the optimal quantization steps computed during the allocation process). An entropy coding is finally applied on the quantized coefficients. In parallel, the connectivity of the coarse mesh is encoded with the coder of [12].

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 at a given specific bitrate R_{target}*:

$$(\mathscr{P}) \begin{cases} \text{minimize} & \sigma_\varepsilon^2(\{q\}) \\ \text{with constraint} & R_T(\{q\}) = R_{target} , \end{cases} \qquad (23)$$

where R_T represents the total bitrate. This constrained allocation problem can be defined by a lagrangian criterion:

$$J_\lambda(\{q\}) = \sigma_\varepsilon^2(\{q\}) + \lambda(R_T(\{q\}) - R_{target}) , \qquad (24)$$

with λ the lagrangian operator. σ_ε^2 and R_T are expressed in function of models for the bitrate and the MSE of each subband. The solutions can be obtained by differentiating (24) with respect to the quantization steps $\{q\}$ and λ, and by solving the resulting system. For more explanations about the coder and the allocation problem, see [11].

To show that the weighted MSE can be exploited for any semiregular mesh independently of the remeshing method, we deal with two versions of this coder. The first version, named the *MAPS* coder, exploits the remesher *MAPS* [5] and the *lifted* Butterfly-based transform. The second version, named the *NORMAL* coder, exploits the *Normal Remesher* [6] and the *unlifted* Butterfly-based transform.

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

We compare the quality of the reconstructed meshes according to two cases: (1) the total distortion is only the sum of the MSE across the subbands; (2) the total distortion is the weighted MSE using the values computed in section 3. The comparison criterion is the frequently used PSNR 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 RMSE between the original irregular mesh and the reconstructed one (computed with *MESH* [13]). Fig. 4 and 5 show respectively the curves PSNR/bitrate (in bits/irregular vertex) for BUNNY encoded with the *MAPS* coder and VENUS encoded with the *NORMAL* coder. We globally observe that using the weighted MSE improves the coding performance for any model and any version of the lifting scheme. In addition, to prove the interest of using the weighted MSE in mesh coding, we compare our coders with the state-of-the-art zerotree coders: PGC [3] including *MAPS* and the *lifted* version of the Butterfly-based transform, and NMC [4] including the *Normal Remesher* and the *unlifted* version. We observe that finally the proposed coders outperform the corresponding zerotree coders (up to 3.5 *dB*).

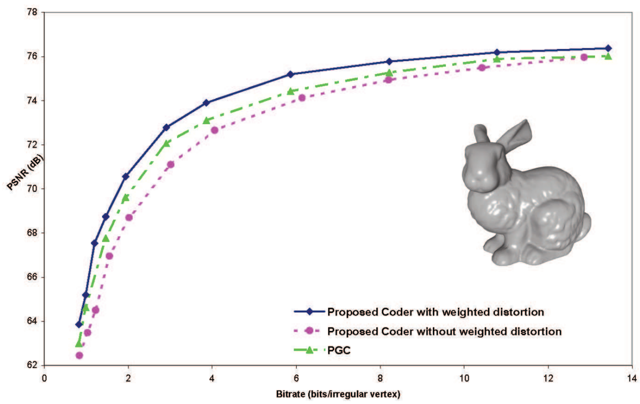

Figure 4: PSNR curves for BUNNY encoded with the *MAPS* coder (*lifted* Butterfly-based transform). $peak = 0.25$.

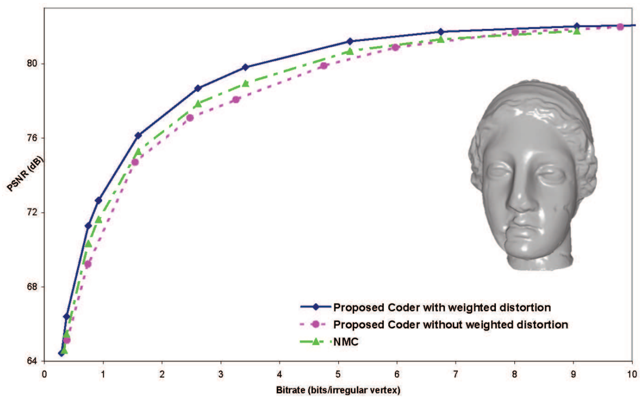

Figure 5: PSNR curves for VENUS encoded with the *NORMAL* coder (*unlifted* Butterfly-based transform) $peak = 1.56$.

5. CONCLUSION

In this paper we have developed the formulation of the weighted MSE on a triangular edge lattice. We show 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 show that using this MSE approximation in a model-based bit allocation improves the coding performance of a wavelet coder for any kind of semiregular meshes and any version of the Butterfly-based lifting scheme: the proposed coders finally outperform the state-of-the-art coders PGC and NMC (up to 3.5 *dB*).

Acknowledgments

Models are courtesy of Cyberware and of Stanford University. We are particularly grateful to Igor Guskov for providing us with his normal meshes.

REFERENCES

[1] K. Park and R.A. Haddad, "Modeling, analysis, and optimum design of quantized M-band filter banks," *IEEE Trans. on Signal Proc.*, vol. 43, n.11, Nov. 1995.

[2] B. Usevitch, "Optimal Bit Allocation for Biorthogonal Wavelet Coding," *IEEE Data Compression Conference*, n.7, Apr. 1996.

[3] A. Khodakovsky, P. Schröder and W. Sweldens, "Progressive Geometry Compression," in *Proc. SIGGRAPH 2000*, 2000.

[4] A. Khodakovsky and I. Guskov, "Normal mesh compression," in *Geometric Modeling for Scientific Visualization* , Springer-Verlag, 2002.

[5] A.W.F Lee, W. Sweldens, P. Schröder, P. Cowsar and D. Dobkin, "MAPS: Multiresolution Adaptive Parameterization of Surfaces," in *Proc. SIGGRAPH'98*, 1998.

[6] I. Guskov, K. Vidimce, W. Sweldens and P. Schröder, "Normal Meshes," in *Proc. SIGGRAPH 2000, Computer Graphics*, Antalya, Turkey, 2000, pp. 95–102.

[7] J. Kovacevic and W. Sweldens, "Wavelet families of increasing order in arbitrary dimensions" *IEEE Trans. on Image Proc.*, 1999.

[8] A. Gersho and R.M. Gray, *Vector quantization and signal compression*, Address: Kluwer Academic Publishers, Norwell, Massachusetts, 1992.

[9] M. Vetterli and J. Kovacevic, *Wavelets and Subband coding*. Address: Prentice Hall PTR, Engelwood Cliffs,1995.

[10] P. Schröder and W. Sweldens, "Spherical Wavelets: Efficiently Representing Functions on the Sphere," in *Proc. SIGGRAPH 95*, 1995, pp. 161–172.

[11] F. Payan and M. Antonini, "Model-based Bit Allocation for Normal Mesh Compression," in *Proc. IEEE international workshop on MultiMedia Signal Processing 2004*, Siena, Italy, Sept. 2005, pp. 327–330.

[12] C. Touma and C. Gotsman, "Triangle Mesh Compression," in *Graphics Interface*, 1998, pp. 26–34.

[13] N. Aspert, D. Santa-Cruz and T. Ebrahimi, "MESH: Measuring Errors between Surfaces using the Hausdorff Distance," in *Proc. IEEE ICME*, vol. 1, 2002.