A Study on Patch-Based Progressive Coding Schemes of Semi-Regular 3D Meshes for Local Wavelet Compression and View-Dependent Transmission

Céline Roudet
UFR Sciences et Techniques
Université de Bourgogne, LE2I
9, Avenue Alain Savary
BP 47870 - Dijon, F-21078
France
Celine.Roudet@u-bourgogne.fr

***ABSTRACT**: This paper firstly introduces a wavelet-based segmentation for three-dimensional (3D) Semi-Regular (SR) meshes, as a pre-processing step, in a region-independent progressive coding algorithm. The proposed segmentation process aims at producing homogeneous regions with respect to their frequency amplitudes on the mesh surface, in other words: patches with different degrees of roughness. We have then studied the behavior of the wavelets, obtained during the independent coding of each region, especially close to the patch boundaries. The main contribution of this paper consists in considering three different possible wavelet decompositions, close to the region borders, and to study their influence during the patch-independent decoding and more particularly during the view-dependent reconstruction process. To our knowledge, no previous work has ever considered the influence of the non-refined parts of a mesh on the more detailed ones, in a view-dependent context. Among the three decompositions we have considered, we defined a new scheme and finally expose various experimentations to demonstrate that it behaves better than the other classical considerations.*

Keywords: Geometric wavelets, Multiresolution analysis, Lifting scheme, Progressive coding, Mesh segmentation, View-dependent reconstruction

Received: 1 September 2010, Revised 3 October 2010, Accepted 9 October 2010

1. Introduction

Three-dimensional (3D) objects and scenes find their way into more and more applications, thanks to the ever increasing bandwidth available in modern telecommunication networks, such as the High Speed Internet, or 3G mobile networks. These kinds of models are not only used in medical applications and entertainment (video games or animation films), but also for industrial purposes like car industry (Computer Aided Design framework), simulation or virtual environment contexts.

The triangular mesh, which is a piecewise approximation of a 3D object shape defined by a set of planar triangular facets, is actually the most common representation for these objects and scenes. This can be explained because it is a simple and effective model for many applications, like rendering for example. It includes geometry (vertex positions) and connectivity data which can be expensive in terms of computation, storage, transmission and rendering, even with today devices and networks.

Consequently in literature, many progressive mesh compression and transmission schemes have been proposed (including [1] - [9]), to represent data with multiple Levels of Detail (LOD). With this representation, a coarse approximation can subse-

quently be further improved depending on the user's devices (network channel and visualization terminal) and expectations. This scalable representation is commonly obtained with a wavelet transform ([4] - [9]), because it produces a good decorrelation of the data. The best compression rates on 3D meshes are currently obtained with a wavelet decomposition, also called Multi-Resolution Analysis (MRA) on Semi-Regular (SR) structures.

MRA transforms a signal into a set of approximations and a set of "details". Hence, it allows to encode not the signal itself, but a coarse approximation (low-frequency signal) and a set of subbands of high frequency details (wavelet coefficients). The information carried by these coefficients is less correlated, that's why it enhances the compression performance. A triangular mesh generally has an irregular connectivity. To our knowledge, only one algorithm [4] directly applies the wavelet transform on irregular meshes, because better compression results can be obtained on SR structures, like in [5] - [9]. A remeshing stage is used to resample the initial models and produce SR meshes, in which the valence of most vertices is equal to six. The cited SR mesh compression methods can hence benefit from a very compact representation of the connectivity information and usually use a zerotree and an entropic coding algorithms [13] to compress the geometry data (regarded as wavelet coefficients).

Most of the state-of-the-art wavelet-based SR mesh compression algorithms ([5] - [8]) consider a global wavelet decomposition on the mesh surface. Given the perpetual increasing expectations of realism imposed by applications and users, increasingly complex geometric models, embedded in the 3D Euclidean space are required. The underlying objects are consequently represented numerically with more and more precision and details. Consequently view-dependent compression, transmission and rendering methods ([9] - [12]) have recently emerged, to accelerate the huge amount of computations involved in this context. The principle consists in considering the user's viewing frustum, to more refine the object regions or scene parts that face toward the viewer. Hoppe [12] was one of the first to introduce the concept of view-dependent simplification and refinement, for fast and efficient rendering of 3D irregular meshes (represented with the Progressive Mesh (PM) [1] structure). Initially, the coarsest mesh is rendered. Then the algorithm iteratively checks whether each vertex need to be further refined, according to the user's viewing frustum.

More recently, several view-dependent compression and transmission algorithms have been developed for SR meshes [9] - [11]. Gioia et al. [10] proposed a wavelet-based view-dependent transmission and visualization algorithm where the wavelet coefficients can be added or suppressed in real-time on a partially reconstructed mesh. Hence the reconstruction of large meshes can be optimized by allocating a major portion of the available bandwidth to visible parts.

Sim et al. developed in 2005 a rate-distortion (R-D) optimized view-dependent SR mesh streaming method [9]. One of the key points is to add a partitioning of the SR mesh into many segments, in order to optimally allocate bits during the compression process. Another advantage is to give the possibility to progressively transmit the segments independently, according to the client's viewing position. This region-based R-D optimization scheme improves the conventional wavelet-based compression algorithms for SR meshes ([5], [6]). The authors considered the same constrained allocation problem as in [7] and [8], to formulate their R-D optimization. Moreover, the server can allocate an adaptive bitrate to each segment (based on its visibility priority), to facilitate the interactive streaming of 3D data. More recently, another view-dependent geometry coding of static 3D scenes (defined by sets of semi-regular meshes) was proposed by Payan et al. [11]. The proposed coding scheme combines a segmentation for determining the visible regions, and an allocation process (detailed in [8]) for improving the visual quality of the encoded scene.

These two previously cited region-based view-dependent coding methods [9] and [11] define two different segmentations: the first one is fixed during the encoding process, whereas the second one is computed for a determined viewing frustum. In this paper, we introduce a new SR mesh segmentation algorithm (adapted from the one detailed in [14]). We also propose to study the behavior of the wavelets in the created regions, during the patch-independent decoding and more particularly during the view-dependent reconstruction process. Our segmentation stage can be seen as a pre-processing step and is not dependent on the user's viewing position. It aims at producing homogeneous regions which share similar frequency amplitudes on the mesh surface, in other words: regions with different degrees of roughness.

None of the two previous algorithms ([9] and [11]) and no other paper (to our knowledge) has ever considered the influence of the non-refined parts of a mesh on the more detailed ones, in a view-dependent context. In the wavelet-based SR mesh compression setting, a subdivision scheme is used as a prediction operator during the MRA, to reduce the details lost during this coarsification process (regarded as wavelet coefficients). The most used prediction scheme for SR meshes is the Butterfly

subdivision stencil [15] (illustrated in Fig. 2 (picture (b)). The main contribution of this paper consists in considering three different possible applications of this stencil, close to the patch borders, and to study their influence during the independent coding of each region. We propose various experimentations to demonstrate that, among these three schemes, the one we have developed can reduce the bandwidth requirement for 3D mesh patch-independent decoding and view-dependent reconstruction.

The remainder of this paper is organized as follows. We begin by an overview of the different parts of our algorithm, followed by an explanation of the MRA on meshes in Section 3, which produces the wavelet coefficients. The wavelet-based segmentation algorithm we developed is then described in Section 4. The different patch-aware applications of the Butterfly prediction are detailed in Section 5. Experimental results are presented in Section 6, together with possible applications of our framework. As a conclusion, discussions and ideas for future work are given in Section 7.

2. Overview of the Proposed Algorithm

Fig. 1 illustrates the main stages of the proposed framework. The algorithm principle is described hereinafter. In this work, an input irregular mesh M_{ir} is first resampled into a SR structure M_{sr}. Our wavelet-based shape partitioning (whose steps are represented in purple in Fig. 1) uses the second generation wavelet formulation (detailed in Section 3). It aims at constructing homogeneous regions, regarding the wavelet coefficient amplitude.

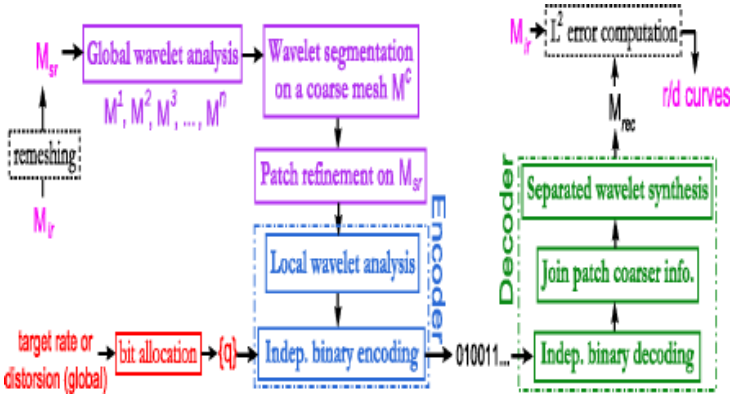


Figure 1. Main features and stages of our framework which aims at locally coding and transmitting a SR mesh M_{sr} (remesh of the original irregular model M_{ir}). Our framework minimizes the reconstruction error for a given bit budget (target rate), with an error-driven wavelet coefficient quantization (q). This allows a flexible reconstruction M_{rec} at the decoder side, where a view-dependent rendering can, for example, be considered. To measure the quality of the reconstructed mesh at a given rate we use rate-distortion (*r/d*) curves (an example is given in Fig. 18)

These coefficients are produced by a global MRA, which decomposes the SR models into n coarser meshes M^1, M^2, ... , M^n and a sequence of "details": the wavelet coefficients. This set of approximation meshes is called hereinafter: "wavelet decomposition hierarchy" or in a simplest way: "hierarchy". Each coefficient, associated to a given resolution level, reveals the high frequencies lost during the mesh simplification.

The goal of our underlying segmentation is to identify the mesh parts where the wavelet coefficients could be further reduced thanks to different decomposition, quantization and/or coding schemes, which are the major steps involved in the progressive compression process. We chose to segment one of the coarsest mesh M^c of the produced "hierarchy".

The segmentation criterion we considered is the averaging of all the wavelet coefficient amplitudes: associated to each mesh of the "hierarchy", starting with M^c (detailed in Section 3.3). Given the segmentation on a coarse resolution level, a fine projection is finally used to partition the finest mesh (the SR original mesh) with the same rule. This pre-processing step gives us the possibility to apply a partition-independent coding and transmission (represented in blue in Fig. 1).

On the synthesis side (represented in green in Fig. 1), after an independent decoding, the coarsest patches are glued and

inversely wavelet transformed. The reconstructed object M_{rec} can finally be compared to the original one, so as to evaluate the distortion.

3. MR Analysis on SR Meshes

In this section, we first detail the theory involved to produce wavelet coefficients on SR structures, with a lifting scheme formulation (second generation wavelets). A SR mesh is obtained thanks to a remeshing stage, which consists in resampling an input model (with irregular connectivity). We present then the concept of the state-of-the-art SR remeshing and MRA algorithms. We finally describe the averaging rule we applied to represent all the details lost during this simplification process, in only one coarse mesh of the "hierarchy".

3.1 Wavelet Theory and Lifting Scheme

The MRA produces a reversible mesh decomposition into a series of approximation meshes and a sequence of details: the wavelet coefficients. Classical MRA methods, such as wavelet transform, were formerly formalized with the filter bank theory. Low-pass and high-pass filters are applied to obtain respectively the approximations and the details. For a discrete signal $s(t)$, its Wavelet Transform is explained by this formulation :

$$WT[s(t)](m, n) = a_0^{-\frac{m}{2}} \int_{-\infty}^{+\infty} s(t)\psi^*(a_0^{-m} t - nb_0)dt \qquad (1)$$

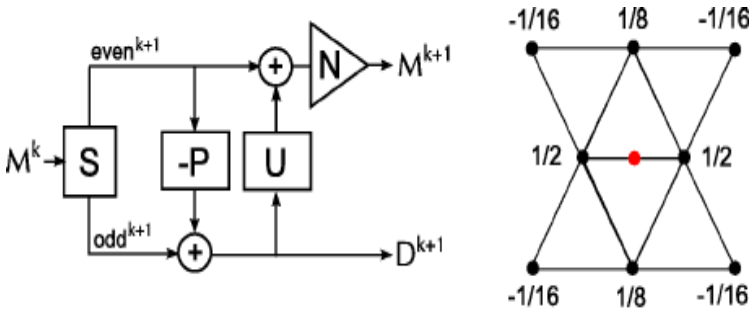
where m, n$\in$ Z and correspond respectively to the scale and the translation parameters (used to analyze the signal in the time/frequency domain). a_0>1, b_0>0 and ψ^* corresponds to the complex conjugation of Ψ.

The choice of a_0 and b_0 produces different tilings of the time-frequency plane. The most used technique is the dyadic analysis, introduced by Mallat [16] in 1989 to link the wavelets and the filter bank theories, where a_0=2 and b_0=1. This MRA formulation, defined for finite energy functions, is formally described for a series of imbricated subspaces $(V^m)_{m\in Z} \subset L^2(R)$ in which an orthonormal scaling function basis is determined. This function family $\{\varphi^{m,n}; n\in Z\}$ is defined by dilations and translations of a mother scaling function $\varphi(t)$ and expressed by the following formula: $\varphi^{m,n}(t) = 2^{-m/2}\,\varphi(2^m t - n)$.

The orthogonal projection of $s(t)$ into the scaling function family forms a set of approximations, obtained by the application of a low-pass filter followed by a uniform subsampling (used in a cascade algorithm launched on the signal $s(t)$). To recover the high frequencies lost after the application of the low-pass filter, we have to take into account the W^i space, which is the orthonormal complements of the V^i space in the V^{m-1} one, for any approximation in the level i. The wavelets, defined by $\psi^{m,n}(t) = 2^{-m/2}\,\varphi(2^m t - n)$, form an orthonormal basis of the Wm spaces and are used as high-pass filtering in order to collect the missed details.

For compression purposes, it is interesting to benefit from wavelets which have at least one vanishing moment, to benefit from a decrease of the wavelet coefficients through the resolution levels. The orthogonality of the wavelets with the scaling functions is also an interesting property because it allows to obtain the best approximations in a least square sense, which is important for visualization and to improve coding performances. This orthogonalization means that $<\varphi^{j,i}, \psi^{j,k}>= 0$ for each pair $(i,k)\in Z^2$ (where <f,g> is defined as the inner product between the functions f and g). Considering the filter bank theory, it's not always possible to construct analysis tools with such properties. Consequently most of the methods have benefited from the lifting scheme formulation, introduced by Sweldens [17]. It moreover reduces computational costs and memory allocation, by first splitting the signal (S operator in Fig. 2, picture (a)) into even and odd components using lazy wavelets. The following lifting steps (P and U operators in Fig. 2, picture (a)) produce a modification of this lazy wavelet basis in order to add the desired properties (we referred previously). These operations are equivalent to a factorization of the pair of the complex filters in the filter bank theory. They can be simplified by the lifting formulation with finite filters. Hence, the multi-resolution synthesis is simply obtained by an inversion of the order and sign of the analysis lifting operators.

The prediction operator (P) is used to predict odd components from even ones, to obtain smaller coefficients by better approaching the higher level mesh than with a canonical quadrisection. The most used prediction operator on SR meshes (the Butterfly scheme [15]) is illustrated in Fig. 2 (picture (b)). The update operator (U) is used to preserve the mean value of the signal and corresponds to the addition of the previously cited properties, like orthogonalization of the scaling functions φ with the wavelets Ψ.



(a) (b)

Figure 2. (a) Principle of the lifting scheme for the decomposition of a fine mesh (M^k) into a coarser one (M^{k-1}). D^{k+1} represents the details lost during the simplification. (b) The most used prediction operator on SR meshes: the Butterfly subdivision stencil [15], which allows to predict the position of each newly added (odd) vertex (in red) thanks to its coarser (even) neighbors

This latter property is applied to minimize the norm of the orthogonal projection of the $\psi^{j,i}$ on the V^j spaces, $\forall\ i,j \in \mathbf{Z}$, with respect to the inner product. Hence the wavelets are expressed by:

$$\psi_i^j = \varphi_i^{j+1} \sum_{k \in \nabla_j} \alpha_{ik} \varphi_k^j \qquad (2)$$

with ∇_j defined as the set of vertices of M^j.

The coefficients α_{ik} of this previous equation are the solution of a linear system which depends on the desired properties. In practice, the orthogonalization is relaxed in order to obtain finite filters. For compression purposes, the decomposition ends up with a normalization step (N), which is used to adjust the scale between the odd and the even coefficients, so as to decompose the surface in a normalized basis.

One of the lifting scheme advantages is that it can be applied to irregular point sets like meshes. In practice, we use a resampling of the irregular meshes (known as remeshing), to regularize the connectivity and to benefit from a more compact representation of the connectivity data.

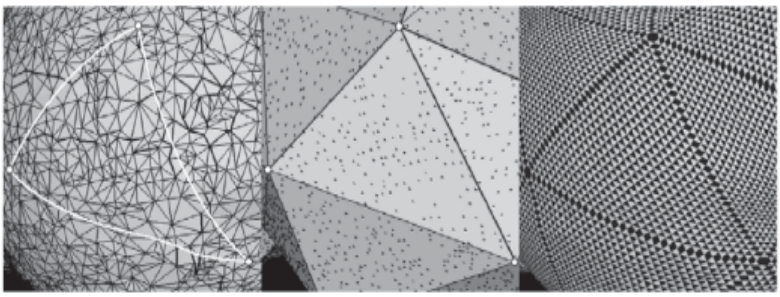
3.2 Remeshing and Wavelets on Meshes

Even though wavelets are commonly used in image or video processing, applying wavelets on triangular meshes is not trivial, because of the irregular sampling of most of them. MR analysis for triangle meshes with arbitrary topology was introduced by Lounsbery [18] who demonstrated that a subdivision scheme can serve as a scaling function basis to extend the wavelet theory for irregular sampled signals like meshes. The compression performances are then highly dependent on the quality of the parameterization defined during the SR remeshing.

Even though wavelets for irregular meshes were proposed in [4], the most popular technique to overcome the problem is the semi-regular remeshing. The principle is to modify the triangulation to make the connectivity as regular as possible. It can be done by resampling the shape geometry, i.e. by moving the vertices onto the surface, while keeping a SR connectivity (constructed with a subdivision scheme). By this way, the majority of vertices has a regular neighborhood (valence is equal to six). It improves the efficiency of the subsequent wavelet analysis on the given triangulated surface, but also highly eases the connectivity encoding, most of which becomes implicit.

As a proof, Khodakovsky *et al*. ([5, 6]) explain that a SR remeshing followed by a wavelet-based compression stage tends to reduce the geometric reconstruction error by a factor four, for one specific compression ratio, when compared to the state of the art connectivity preserving methods. Hence, this is not surprising that the best mesh coders are the wavelet-based coders for SR meshes ([5] - [9]).

All the SR remeshers for triangulated surfaces are based on the same idea. They aim at reducing the parametric information (tangential set of wavelet coefficients), to describe the surface with only the geometric information. For that purpose, a mesh simplification is used to produce a coarse mesh (called the base complex) on which the input model is parameterized. The principle (illustrated by Fig. 3) is the following: *i*) an irregular mesh M_{ir} is first simplified to obtain a coarse base complex S_0, on which the original vertices of M_{ir} are projected during the parameterization stage; *ii*) each triangle of S_0 is then subdivided by triangle quadrisection, and each new regular vertex added by subdivision is moved on the original surface by using the parameterization previously determined.



(a) (b) (c)

Figure 3. Overview of a semi-regular remeshing. Each vertex in the area bounded by white curves (a) is projected in one triangle of the base complex (black dots on (b)). The SR mesh (c) is then obtained by iterative subdivision and by mapping the new vertices onto the surface using the parameterization. Image of [9]

By iterating several times this second step, a set of more and more detailed approximations S_0, S_1, S_2, ... is created. This refinement step is stopped when the surface modeled by an approximation is fine enough to be geometrically or visually similar to the surface modeled by M_{ir}. The resulting approximation is the output SR mesh M_{sr}. Even though the two meshes M_{ir} and M_{sr} represent the same surface, a remeshing error remains, obviously negligible if the remeshing stage is efficient.

During the simplification stage, two approaches are generally proposed to obtain the base complex with a topology similar to the original shape: either by constructing on the initial object a mesh chartification followed by a patch simplification [19, 20, 21, 22] or by applying topology-preserved progressive decimations [23, 24, 25]. During this step, a parameterization function is defined to minimize the distortion when mapping each curved surface to its associated coarse triangle in the base complex. The challenge consists in minimizing the distortion produced during the parameterization of the input model, when mapping each curved surface to its associated coarse triangle in the base complex.

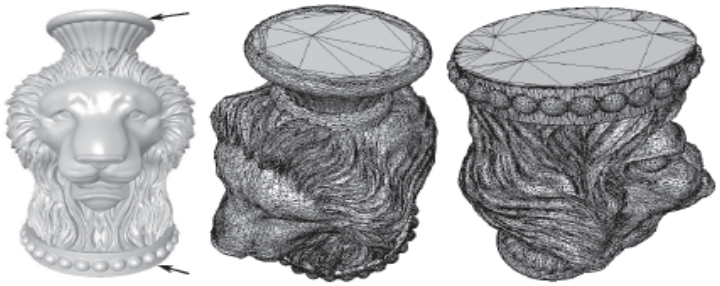
The Normal Mesh algorithm [24] we considered to obtain most of the SR models presented in this work belongs to the second class and uses iterative half-edge collapses to decimate the vertices. The main contribution of this method lies in the refinement step. The authors first propose to subdivide the triangles of the base complex using the interpolating Butterfly scheme [15, 26]. Once positioned with this technique, the new vertices are then moved along their normal direction until they "pierce" the original surface. This refinement step is repeated several times until we get a final SR output with low distortion, called a Normal Mesh.

The Normal Meshes are particularly relevant for compression ([27, 6, 8]). Indeed, if the unlifted Butterfly wavelet transform is applied during analysis, the wavelet coefficients produced are guaranteed to be in the local normal directions. Therefore the x and y coordinates (*i.e*. the tangential components) of the coefficients are equal to zero, and the geometry information is fully represented by the z coordinate (*i.e*. the normal component). Hence, only one single scalar per vertex must be encoded instead of three. That's why this algorithm currently produces one of the best remeshes for compression purposes, but only for closed surfaces which have uniform triangles.

Since the Lion Head original model has non uniform triangles (as illustrated in *Fig. 4*), we considered another remeshing algorithm to obtain a good SR model. One of the most recent SR remesher, developed in 2010 by Kammoun *et al*. [22], includes a pre-processing step to resample the irregular mesh and solve the problem of non uniform triangles. Their SR remesher belongs to the first class and creates a coarse Voronoï partition on the original mesh in order to simplify it and form the base

complex with the dual construction: the Delaunay triangulation. At the same time, a parameterization function is defined to minimize the distortion when mapping each curved surface to its associated coarse triangle in the base complex and is used to refine it after applying a subdivision step.

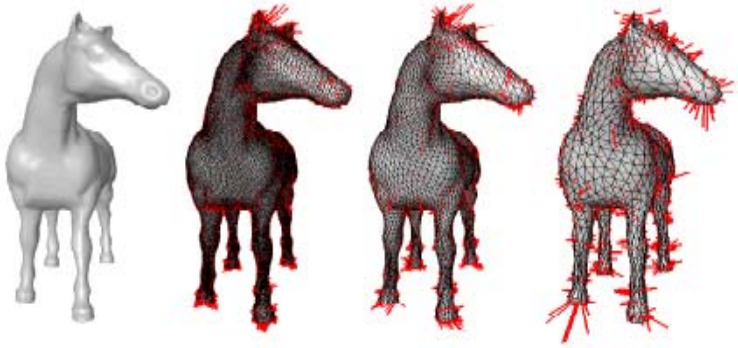
Given SR meshes with the fewest connectivity and parametric data to encode, efficient data structures and processing algorithms are used to get closer to the methods used for data sampled on regular grids. The application of the MRA on SR meshes, resulting from these latter algorithms, and mainly used for progressive compression purposes, can be based on various subdivision schemes. Most of the existing methods have benefited from interpolating subdivision filters (and in the most cases from the Butterfly stencil) for the low-resolution versions to be good approximations of the original object (in a least-squares sense). In other words, to provide numerical stability of the fitting operation and to have a more stable wavelet construction for practical applications than with approximating schemes.



(a) (b) (c)

Figure 4. (a) Original irregular Lion Head model. Other views are proposed to better appreciate the non-uniform triangulation, where big triangles can be seen on the top (b) and on the bottom (c) of the head

Each mesh refinement, processed during the MR synthesis, is based on canonical quadrisections applied on every facet. This one-to-four triangle construction consists in adding three new vertices in the middle of each coarse edge. The prediction operation, followed by the wavelet coefficient addition, enables to recover the real position of these newly added vertices. Consequently the common representation of the wavelets consists in associating them with their corresponding coarse edges, as we can see in *Fig. 5*, on the first three decompositions of the Horse model.



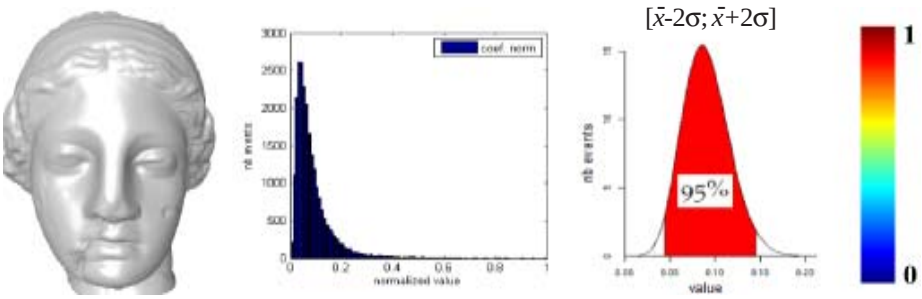
(a) (b) (c) (d)

Figure 5. (a) Original Normal Horse model; (b-d) Wavelet coefficients represented as 3D vectors (in red) linked to the edges of the first three decompositions (multiplication factor used to emphasize them: 10)

Most of the actual state-of-the art mesh compression methods ([5] - [9]) apply a global wavelet decomposition (same lifting filters on the entire surface). Since the considered prediction schemes generate smooth finite surfaces, another kind of prediction, quantization and/or coding could be examined for the non-smooth parts of the mesh. We developed our segmentation process in this sense and considered only Normal Meshes in our study, since they have been proven to provide the best compression rates.

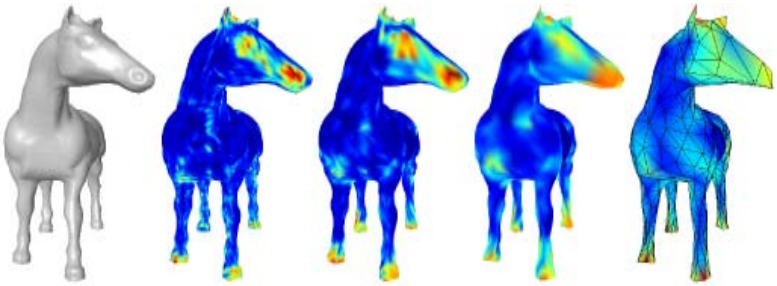
3.3 Wavelet Coefficients seen as a Segmentation Criterion

We propose to study the wavelet coefficient amplitude distribution in the "hierarchy" produced by a global MRA. Our goal is to create regions on the mesh surface with different degrees of roughness: reflected by the wavelet coefficient amplitude. We illustrate in *Fig. 6, 7 and 8* the normalized distribution of the coefficient amplitude, on various resolution levels of the Horse and Venus models (remeshed by the Normal Mesh [24] algorithm).



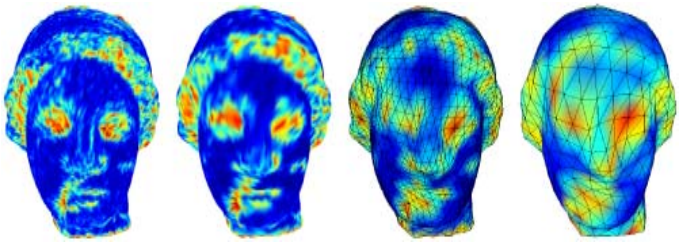
(a) (b) (c) (d)

Figure 6. Distribution of the wavelet coefficient norm, on the first resolution level of the Normal Venus model. (a) Original model; (b) Histogram of the normalized coefficient norms associated to every resolution level; (c) The Gaussian normalization method we used to define a confidence interval (in red); (d) Color scale used in this figure and the next ones



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

Figure 7. Gaussian normalized distributions of the wavelet coefficient norm, on the first four resolution levels of the Normal Horse model. 7 Gaussian normalized distributions of the wavelet coefficient norm, on the first four resolution levels of the Normal Horse model



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

Figure 8. Gaussian normalized distributions of the wavelet coefficient norm, on the first four resolution levels of the Normal Venus model

We applied a Gaussian normalization on the amplitude distributions (represented in *Fig. 6, picture (c)*), whose goal is to narrow extreme values. It hence allows to easier classify and segment a mesh with the considered criterion, since we can retain only the most represented samples. The application of this statistical technique (known as a confidence interval) is possible because the related histogram (depicted for the Venus model in *Fig. 6, picture (b)*) has a Gaussian-like normal distribution. Given the distribution averaging ($\bar{x}$) and the standard deviation (σ), 95% of the population is kept in the chosen confidence interval.

On the globally smooth models we studied, the coefficient amplitude distributions allow to identify high curvatures characterizing the eyes, nose, mouth or ears, and also the object tips (neck or feet). This spreading emphasizes also the textured parts, such as the hair of the Venus head. These experimental amplitude distributions corroborate our previous hypothesis about the faculty of the subdivision surfaces to well predict the mesh smooth parts. Moreover even in globally smooth models, there remain some areas where other considerations should be employed.

In the Horse and Venus first four resolution levels (*Fig. 7 and 8*), the same main features are underlined, but the fineness of these highlighted high frequencies is quite different in each level. Since we aim at producing a segmentation that identifies clearly the high frequencies lost during the coarsification process, a global consideration of all the coefficient amplitudes in the entire "hierarchy" is required. The segmentation algorithm we used in this context was conceived to apply on one mesh at a time. Consequently instead of segmenting all the approximations, we propose to compute an averaging of all the wavelet coefficient amplitudes.

We chose to segment one of the coarsest resolution levels, for each coarse triangle to be assigned a unique region number. It gives us the opportunity to independently treat each produced region, without being forced to remesh them, in the case where a subdivision connectivity is not encountered. Hence the chosen coarse level determines the number of resolution levels involved in the patch-independent MRA. Considering this setting, the averaging rule we adopted is illustrated in *Fig. 9*. Each edge of the coarse level is then associated to two children edges (called *fae* in *Fig. 9*) in the immediately finer mesh. We applied a weighted averaging where each coefficient norm is multiplied by 2^d, to take into account the decrease of the wavelets in the "hierarchy" (explained in *Section 3.1*) - *d=0* for the coarsest edges and increasing integer values for the finer decompositions.

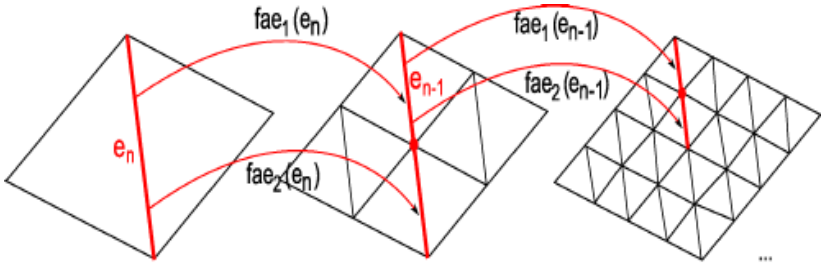


Figure 9. Wavelet "fine averaging association". For an edge e_n in a coarse decomposition, we compute an averaging of its wavelet norm and the norms of its finer associated edges (*fae*) in all the existing resolution levels

4. Wavelet-based Classification and Segmentation

Given the wavelet norms, averaged on one of the coarsest decompositions, we used an extension of the algorithm developed by Lavoué *et al.* [28], to create mesh segmentation. This method is based on a K-Mean classification [29].

4.1 Mesh Classification and Segmentation Algorithms

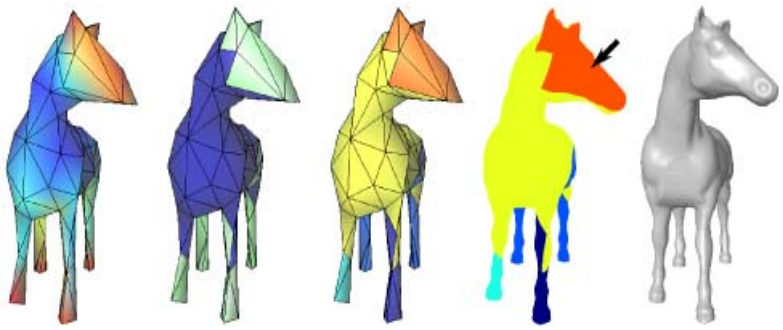
In the past twenty years, much work has been done in shape decomposition because of its utility in many computer graphic applications. These methods are usually designed to solve specific problems and are difficult to compare. Some applications use segmentation principles to simplify treatments like texture mapping, parameterization, mesh editing, deformation or compression on complex meshes with a high genus, for example. They are generally faster and more simple on surface patches homeomorphic to a disc. The partitioning criterion generally used in this setting is the discrete curvature or the planarity information. Our segmentation process uses a different criterion together with an adaptation of the classification and segmentation algorithms implemented by Lavoué *et al.* [28].

They used a curvature tensor estimation [30] to compute the principal curvature values in each vertex. The produced segmentations delimit smooth regions surrounded by sharp edges on CAD objects. Our adaptation replaces this curvature criterion by our previously defined wavelet norm averaging.

The K-Mean classification algorithm was first used to create two groups of vertices (referred as clusters): one with the smallest amplitudes (represented in dark blue in next figures, and referred as the smooth cluster) and the other with the highest ones (in light green). After several iterations, starting with two randomly determined centroïds, each vertex is then associated to a cluster C_i and a "roughness value" r_i. Homogeneous connex patches are then deduced with a region growing and labeling algorithm adapted from [28]. More precisely, the studied measure is transmitted from vertices to triangles, starting from seed triangles whose incident three vertices belong to the same cluster. Thanks to this framework, a mesh decomposition in a small number of regions can be created at any given resolution level.

4.2 Segmentation results

We illustrate in *Fig. 10 to 12* the different steps involved in our segmentation process, respectively for the Horse (*113K vertices*), Venus (*164K vertices*) and Skull (*131K vertices*) Normal models and also for the Lion Head model (*676K vertices*), remeshed uniformly by the recent algorithm developed by Kammoun *et al.* [22]. Colors used for the patches illustrated in *Fig. 10 and 11* were randomly chosen. The arrows in the pictures serve to indicate which region will be further studied in the rest of the paper. The result of our averaging rule is depicted on meshes which have a various number of vertices (because different numbers of resolution levels were considered). It has a tendency to stump the isolated details (which can be considered as local geometric noise) and sharpen the very important ones (highlighted in various resolutions and close to the mesh saliency concept of Lee *et al.* [31]). The number of created patches is fully dependent on the choice of the coarse level. We examined different choices, to study the influence of the chosen coarse level, on the further detailed experimentations.



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

Figure 10. Illustration of the different steps involved in our classification and segmentation processes, on the 5^{th} level of the Normal Horse model. (a) Averaging of the wavelet norms in the 5^{th} level followed by (b) the K-Mean classification and (c) segmentation in 5 regions; (d) Fine projection of the segmentation on the SR original model whose features can be better appreciated with a Phong shading (e)

Given the segmentations of the Normal models on a coarse level, the final stage of our pre-processing consists in projecting the coarse regions on the original SR model. This stage is simply realized incrementally, resolution per resolution, where the four children facets associated to each coarse triangle are assigned the same cluster and region number as their father triangle. The resulting fine segmentations can now be the input of our patch-independent MRA, detailed in the following.

5. Region-based MRA for Patch-Independent Reconstruction

As a preliminary step of the development of a locally-based R-D optimized coding scheme, we propose to detail in this section the three different applications of the Butterfly stencil we examined close to the patch borders, during the region-based MRA and coding. In the next section, experiments are presented to study the influence of the non-refined parts of a mesh on the more detailed ones, considering each of these three rules for patch-independent decoding and more specifically in a view-dependent reconstruction context.

5.1 Butterfly stencil adaptations for region-based MRA

In our first consideration (designed later by "No PB": for No Patch Borders taken into account), the unlifted Butterfly wavelet decomposition is realized as if any patch was defined. This rule is accurate if each region is quantized and/or reconstructed uniformly (with approximately the same L^2 error orders, in each region). In other cases, an alteration of the quality of some

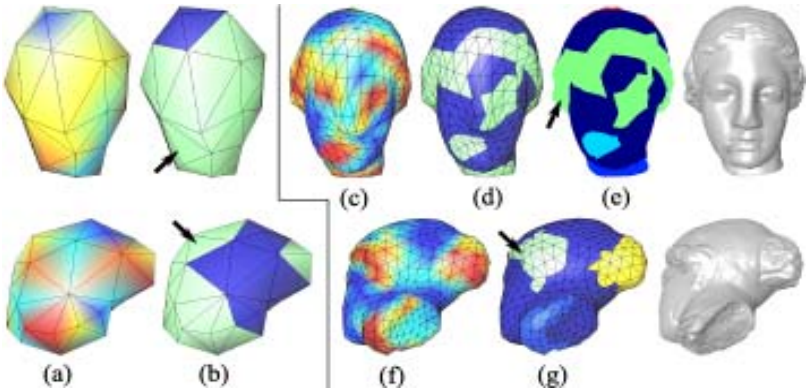


Figure 11. Illustration of the different steps involved in our classification and segmentation processes, on the 6th (left) and the 4th (right) levels of the Normal Venus model. (a, c, f) Averaging of the wavelet norms in a coarse level followed by (b, d) the K-Mean classification and (g) segmentation in 6 regions (on the 4th level). Three regions are obtained in the 6th decomposition (left). (e) Fine projection of the 4th level segmentation on the SR original model. 11 Illustration of the different steps involved in our classification and segmentation processes, on the 6 (left) and the 4 (right) levels of the Normal Venus model. (a, c, f) Averaging of the wavelet norms in a coarse level followed by (b, d) the K-Mean classification and (g) segmentation in 6 regions (on the 4 level). Three regions are obtained in the 6 decomposition (left). (e) Fine projection of the 4 level segmentation on the SR original model

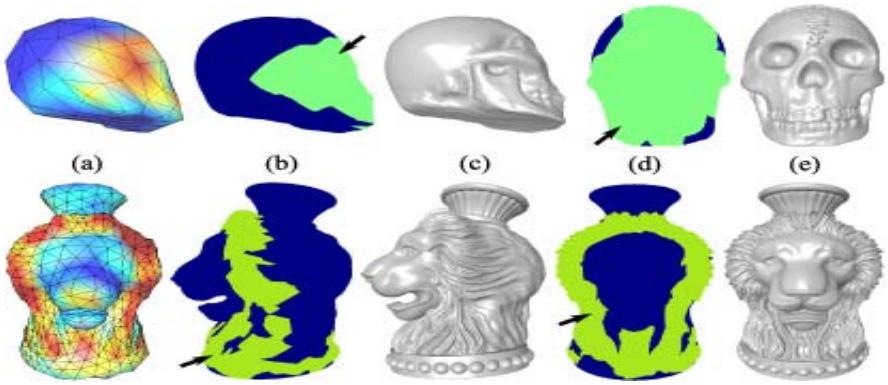


Figure 12. Illustration of the different steps involved in our classification and segmentation processes, on the 5th level of the Normal Skull model (top) and of the Lion Head model (bottom). (a) Averaging of the wavelet norms in the 5th level. The non-smooth region is depicted in green in the SR original model (b, d), whose features can be better appreciated with a Phong shading (c, e)

regions can appear due to the influence of the "poor" quantization or reconstruction of their neighbors, for example.

For these other cases (which don't necessarily involve the same kind of quantization and reconstruction in each patch, like adaptive quantization or view-dependent reconstruction), we propose to examine two different adaptations of the Butterfly stencil close to the patch borders. These two adaptations (designed later by "PB") will be applied when each patch is seen as an independent non-closed surface. With this latter setting, any region can influence its neighbors during the "non-uniform" region-based reconstructions.

Fig. 13 shows the amplitude and polar angle (the angle between each coefficient and its related surface normal vector) distributions for each of the three rules. We can deduce that the two predictions associated to the "PB" consideration (histograms on the middle and on the bottom) are not as good as the classical ("No PB") one (on the top), because the wavelet amplitude and polar angle distributions are generally more dispersed. The first rule we considered (close to the patch

borders) uses two extensions of the classical Butterfly stencil (illustrated in *Fig. 14*) to predict the new vertices situated on the borders (*picture (a)*) or surrounded by extraordinary vertices (*picture (b)*), as it can appear close to the patch borders.

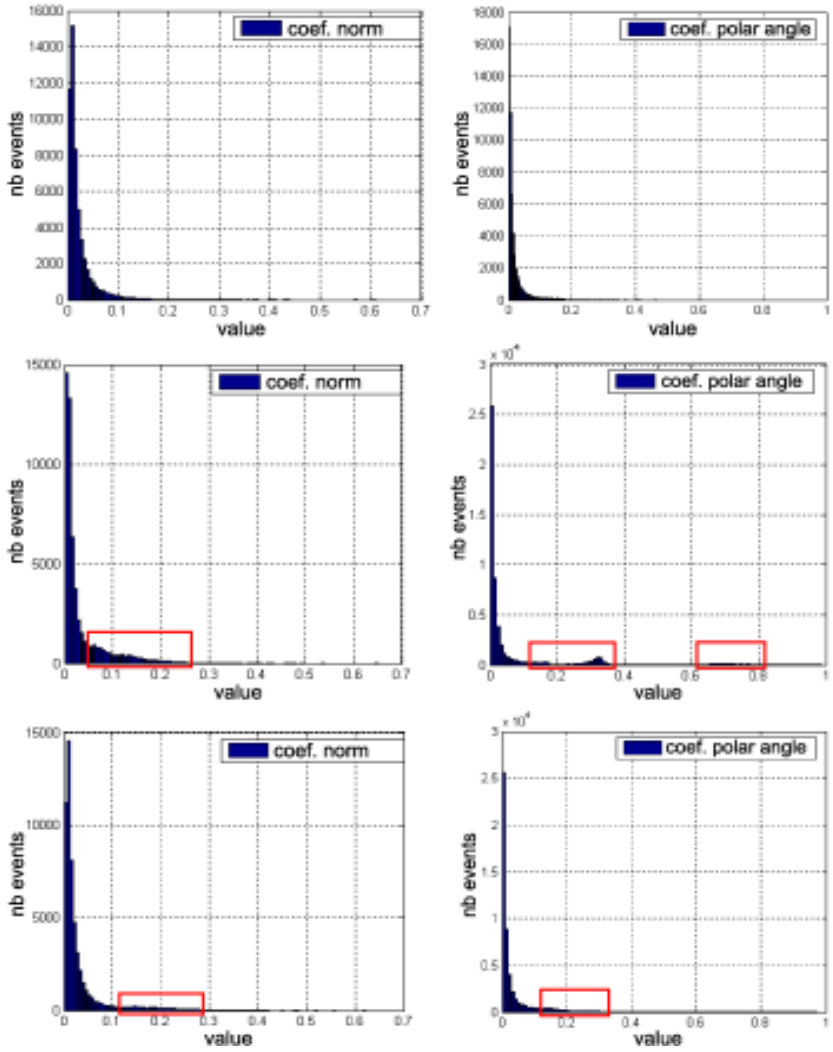


Figure 13. Amplitude (left) and polar angle (right) distributions in every resolution level of the Venus Normal segmented model (right segmentation of *Fig. 11* considered). The red boxes highlight the differences noticed in the two “PB” distributions (middle: classical rule - bottom: our new rule), in comparison with the distributions for which the classical Butterfly stencil was used, without taking into account the patch borders (top)

The goal of our new rule (designed later by "PB SYM") is to recreate the "missing coarse vertices" in the classical Butterfly stencil (illustrated in *Fig. 2*). These vertices are missing because we take into account the patch borders in the MRA. The position of these created "ghost vertices" is deduced from the other real coarse vertices involved in the Butterfly stencil. We illustrate in detail in *Fig. 15* the concept we used for deducing them, with a tendency to favor the diagonal symmetry. We also enumerate all the possible cases we can encounter and for the ones for which a diagonal symmetry cannot be computed, we define an equivalent rule (illustrated with colors in the middle of *Fig. 15*).

In *Fig. 13*, we can observe that our new rule distributions (on the bottom) are closer to the classical ("No PB") ones (on the top). Moreover we have the possibility to perfectly decode some regions of a mesh while other are at the same time "poorly" reconstructed.

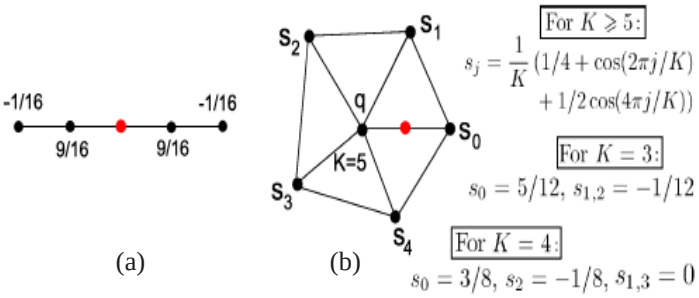


Figure 14. (a) Special Butterfly stencil used to predict the red vertex, situated on the mesh border (for non-closed 3D meshes); (b) Extension of the classical Butterfly stencil for extraordinary vertices (scheme proposed by Zorin et al. [26])

5.2 Patch-Independent Wavelet Encoding and R-D optimization

Fig. 16 presents the main steps of a wavelet-based coder for SR meshes. The principle is the following: i) the original irregular surface M_{ir} is semi-regularly remeshed; *ii*) the wavelet transform is applied to decompose the SR mesh M_{sr} until we obtain the coarsest approximation (which represents the "low-frequency" subband); during this coarsification, many high-frequency details (the 3D wavelet coefficient subbands) are lost and must be kept to allow the reconstruction of the mesh; *iii*) concerning the high-frequency subbands, we first quantize them and then apply an entropic encoder (Huffman, arithmetic, ...) to obtain a binary information; a scalar quantization is considered, so there are indeed three independent encoders (one for each coefficient coordinate); *iv*) the coarse mesh connectivity is then losslessly encoded separately with a single rate encoder (like the Touma and Gotsman [32] one).

For wavelet coefficient quantization, we considered the Rate-Distortion (R-D) optimized coding scheme developed by Payan *et al.* [8] and based on a Lagrangian optimization. This algorithm minimizes the reconstruction error for a given bit budget, with an error-driven wavelet coefficient quantization. More specifically, we used their adaptation for region-based wavelet-encoding [11] so as to optimally quantize the wavelets and allocate the bits in each partition, considering each region distortion contribution with respect to the entire mesh surface. In this adaptation, little additional information need to be added: the number of created partitions, the coarse vertices situated in the borders, each partition cluster affiliation and the optimized quantization steps computed for each region.

We also used the EBCOT encoding Payan *et al*. have adapted for SR meshes [8], before applying the arithmetic encoding. It allows to reduce the intra and inter-resolution correlation. Combined with their R-D optimization, it improves the encoding performance up to +2.5 dB in comparison with the original zerotree coder of Khodakovsky *et al*. [5].

Given all these encoding treatments, the data flow is now ready to be transmitted over the network and reconstructed on the client's side. The analysis stages we have just described are reversed during the partition-based MR synthesis (and detailed in *Fig. 1*). It produces a more flexible reconstruction than with a global decoding.

6. Experimental Results and Applications

We present in this section, various experimental results we obtained with our application implemented in C++. It uses the Computational Geometry Algorithm Library (CGAL [34]) and more specifically the polyhedral surface package. These experimentations were obtained on the meshes we segmented with our wavelet-based partitioning algorithm, but any other kind of SR mesh segmentation could also have been considered, like the ones implemented by Sim *et al.* [9] or by Payan *et al*. [11].

We first propose a visual comparison (in *Fig. 17*) of the mesh distortion involved when we consider different wavelet decompositions close to the patch borders (detailed in the *Section 5.1*). Indeed the compression scheme we used (illustrated in *Fig. 16*) produces reconstructed meshes with a distortion, in comparison with the original irregular mesh. First an

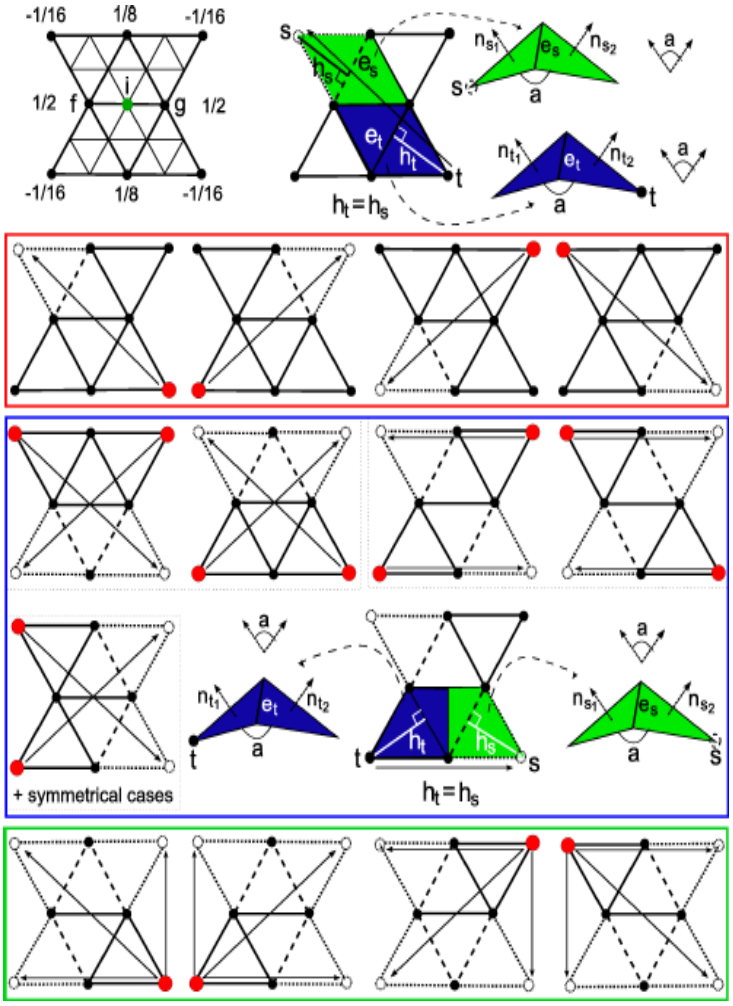


Figure 15. Definition of our new scheme, used when some of the coarse vertices are missing in the Butterfly stencil, due to patch borders (illustrated with big dashed lines). Top left: classical Butterffly stencil; Top right: method used to recreate the "missing coarse vertices" in the Butterfly stencil. We favor the diagonal symmetry (computed thanks to the red vertices in the next pictures), but for the cases where it is not possible, we define an equivalent rule (illustrated with colors in the middle of this figure). The different configurations are highlighted with big rectangles, when only a coarse vertex (top in red), two coarse vertices (middle in blue) or three coarse vertices (bottom in green) are missing

approximation error is involved during the remeshing step, moreover the geometric coding introduces also losses during the quantization stage. State-of-the-art mesh compression methods generally interpret this distortion as a geometric distance between the original irregular mesh M_{ir} and the compressed SR mesh $\hat{\boldsymbol{M}}_{sr}$.

The Mean Square Error (*MSE*) for meshes (inspired from the *Hausdorff* distance) is generally used to evaluate this geometric distance. It corresponds to an average of the point to surface distance between two surfaces *X* and *Y*, given by this formula:

$$d(X,Y) = \left(\frac{1}{area(X)} \int_{x \in X} d(x,Y)^2 \, dx\right)^{\frac{1}{2}} \qquad (3)$$

where $d(x,Y)$ is the Euclidean distance from a point x on X to the closest point on *Y*. Since this L^2 distance is not symmetric, we symmetrize it by taking the max of $d(X,Y)$ and $d(Y,X)$. In our context the *MSE* is hence given by the following equation

(computed with the *Mesh* software [36]):

$$MSE = max(d(M, \hat{M}_{sr}), d(\hat{M}_{sr}, M)) \quad (4)$$

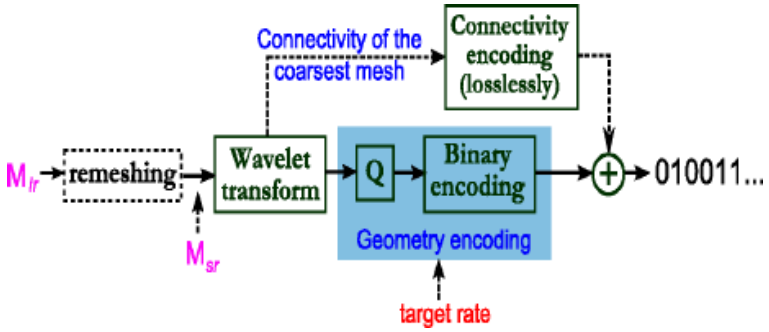


Figure 16. Main steps of a wavelet-based coder for SR meshes. Q stands for the Quantization step. The geometry encoding is generally done under a constraint which can be user-defined or automatic. This parameter can be a file size [5], a rate [8, 33], a reconstruction quality, etc

In *Fig. 17* we compare two different "view-dependent" reconstructions of the segmented Skull model (produced by the "No PB" consideration). Hence we computed the *MSE* with the Mesh software [36], on the entire Skull model (in comparison with the irregular original model), and also on its non-smooth cluster. In the left part, any prediction operator nor wavelets (only canonical quadrisections) were considered on the smooth cluster. Hence it can be interpreted as a non-visible part in a view-dependent consideration, whereas the other cluster is perfectly decoded. The only difference in the right part is that the Butterfly prediction was used in the smooth cluster. As expected, these pictures show that a less important distortion can be obtained with the Butterfly prediction, when patch borders are not taken into account during the MRA ("No PB").

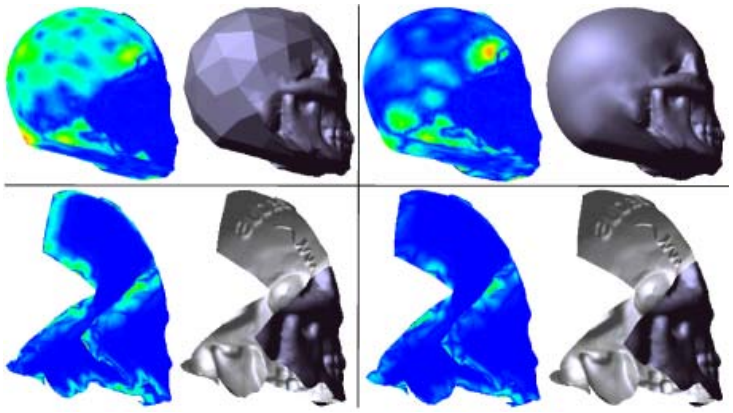


Figure 17. Mesh distortion on two different "view-dependent" reconstructions, presented on the entire mesh (top) and on the non-smooth cluster (bottom) of the Skull segmented model (illustrated in *Fig. 12*). Patch borders are not taken into account during the MRA (referred as the "No PB" consideration) and any wavelets are added on the smooth cluster. In the left pictures any prediction operator is considered, whereas in the right ones the Butterfly prediction is used. The colors depend on the magnitude of the local *MSE*, from the minimal (*blue*) to the maximal (*red*) value. These pictures were realized thanks to the Mesh software [36].

This assumption is also confirmed by the *Table 1*. It results from a computation of the *MSE* for the "No PB" and "PB" considerations, with or without a Butterfly prediction on the smooth region(s). The distortion is presented for all the

segmented models we previously commented, only for one of their non-smooth regions (indicated with an arrow in *Fig. 10 to 12*). In each studied mesh region, the smallest L^2 errors (*MSE*) are emphasized with bold characters. The same errors are obtained in the last two columns, because we used a "PB" consideration which doesn't depend on the prediction used in the neighboring smooth cluster and thus gives the smallest distortions. For each model, the "No PB" consideration associated with a Butterfly prediction in the smooth cluster always gives the better results than without any prediction.

	nb lev	# Vsr	No PB	No PB Butt	PB	PB Butt
Venus part.	6	21,545	$9.053.10^{-4}$	$4.107.10^{-4}$	$8.301.10^{-7}$	$8.301.10^{-7}$
Venus part.	4	76,193	$1.345.10^{-4}$	$1.015.10^{-4}$	$1.456.10^{-6}$	$1.456.10^{-6}$
Horse part.	5	7,857	$6.414.10^{-5}$	$5.616.10^{-5}$	$7.775.10^{-8}$	$7.775.10^{-8}$
Skull part.	5	56,865	$5.260.10^{-4}$	$3.433.10^{-4}$	$5.990.10^{-7}$	$5.990.10^{-7}$
Lion Head part.	5	239,054	$2.094.10^{-4}$	$1.582.10^{-4}$	$7.630.10^{-7}$	$7.630.10^{-7}$

'nb lev' = number of resolution levels.
'# V_{sr}' = number of vertices in the SR mesh.
'Butt' = the Butterfly prediction was used in the smooth cluster.

Table 1. L^2 errors for the "No PB" and "PB" considerations, on the regions indicated with an arrow in *Fig. 10 to 12*. No prediction operator or the Butterfly prediction and no wavelets were considered on the smooth cluster, whereas the other cluster was decoded perfectly. 1 L errors for the NoPB and PB considerations, on the regions indicated with an arrow in Fig. 10 to 12. Any prediction operator or the Butterfly prediction together with any wavelets were considered on the smooth cluster, whereas the other cluster was decoded perfectly

We finally present the bitrate-*PSNR* curves in *Fig. 18 and 19*, computed by only taking into account the non-smooth region of the segmented Skull and Lion Head models (the Butterfly prediction is used -without any wavelets- in the smooth region). The bitrate is reported with respect to the number of vertices in the semi-regular (SR) mesh. These curves were obtained with the view-dependent geometry coding (developed by Payan *et al.* [11]), which we adapted to our segmentation process.

For a comparison of the compression performances on objects with different sizes, we use the *PSNR* (Peak Signal to Noise Ratio) measured in decibels (*dB*). It computes the ratio between the signal dynamic and the reconstruction error: the lower the error is, the higher the *PSNR* is. The *PSNR* for triangulations is given by the following formula:

$$PSNR = 20.\log_{10}\frac{BBdiag}{MSE} \qquad (5)$$

where *BBdiag* is the original mesh Bounding Box diagonal.

The results we deduced from the *Table 1* are confirmed by the repartition of the curves (the "PB" consideration gives the best results at any bitrate). Moreover we can notice that our new prediction scheme (associated to the "PB SYM" consideration) performs better than the other "PB" rule, for patch-independent reconstruction purposes. For the "No PB" consideration, we can see that the two curves in *Fig. 18* reach quickly their asymptotic value which is linked to the L^2 errors illustrated in the bottom part of the *Fig. 17*.

For image and video lossy compression, the *MSE* is most often chosen as a quality measure because it is well adapted to these data and reflects the visual quality even at low compression rates. But it is more complex for 3D meshes, since a distortion metric cannot consider only the mesh geometry, because the visual quality also depends on the mesh connectivity, curvature, smoothing, ... *Fig. 20* illustrates this problematic where the two objects have the same *MSE* (e_1), but the first reconstruction (in the middle) better preserves the sharp features (feet and head of the Horse model). This latter reconstruction has been produced by our framework, where a more important number of bits were allocated in the identified "non-smooth" regions than for the rest of the body. On the other hand, the second reconstruction (on the right) was produced by a global encoding of the mesh. The Horse head was zoomed in, to better appreciate it.

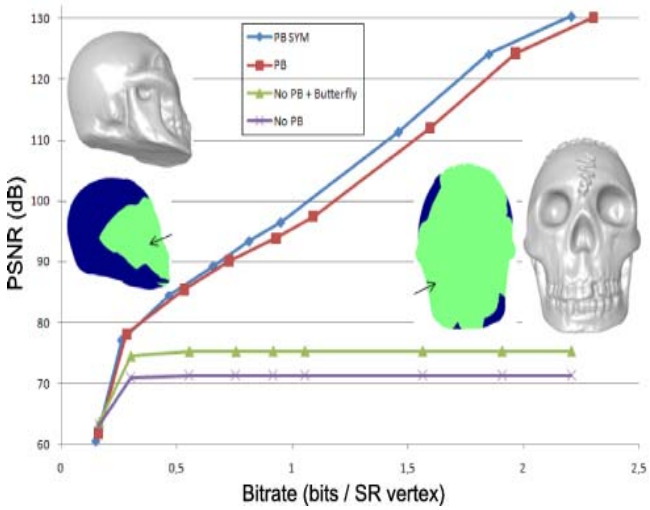


Figure 18. Bitrate-*PSNR* curves for the non-smooth region of the segmented Skull model, where the Butterfly prediction is used (without any wavelets) in the smooth region. The bitrate (reported in bits/SR vertex) corresponds to the bit allocation after quantization, with respect to the number of vertices in the SR non-smooth patch. 18 Bitrate-PSNR curves for the non-smooth region of the segmented Skull model, where the Butterfly prediction is used (without any wavelets) in the smooth region. The bitrate (reported in bits/SR vertex) corresponds to the bit allocation after quantization, with respect to the number of vertices in the SR non-smooth patch

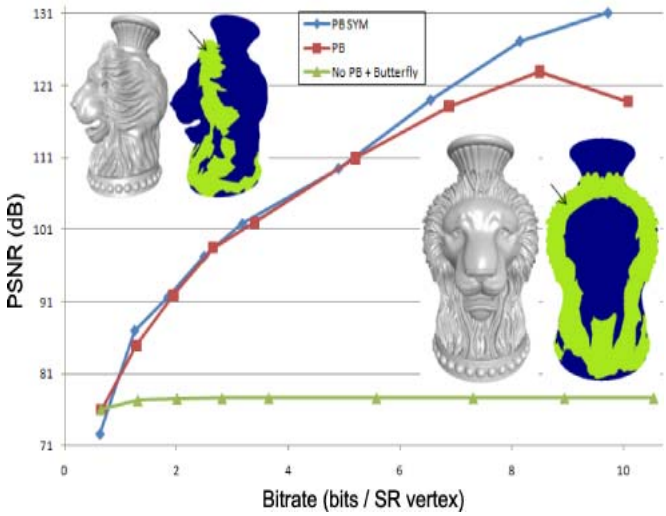


Figure 19. Bitrate-*PSNR* curves for the non-smooth region of the segmented Lion Head model, where the Butterfly prediction is used (without any wavelets) in the smooth region. The bitrate (reported in bits/SR vertex) corresponds to the bit allocation after quantization, with respect to the number of vertices in the SR non-smooth patch

In the figure we also mention a second distortion measure (e_2), called MSDM and introduced by Lavoué et al [37] in 2006. MSDM uses geometric metrics, based on curvature analysis computed on local spherical windows in the analyzed mesh. Similar objects have a distortion close to zero. The score obtained by our patch-based reconstruction seems to better reflect the perceptual distance, the *MSE* cannot reveal. In 2010, Benhabiles et al [38] have proposed an extensive experimental comparison of existing similarity metrics for 3D meshes. This comparative study includes a subjective experiment with human

observers, to evaluate these metrics according to several criteria.

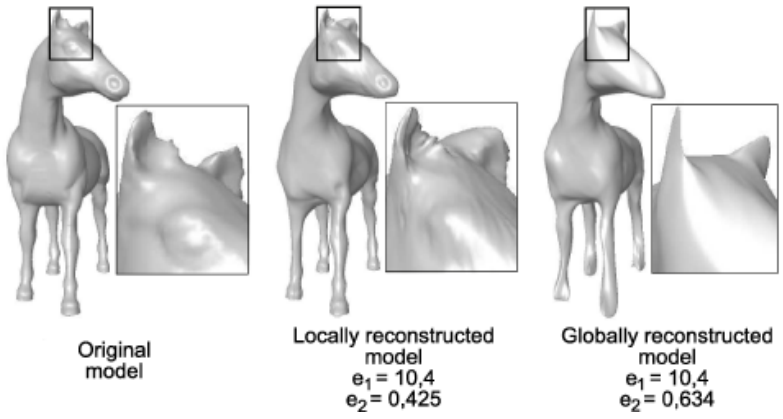


Figure 20. Comparison of two different reconstructions of the Horse Normal model based on a local (middle) and a global (right) wavelet decomposition. The reconstruction presented in the middle, was produced with our framework on the segmented Horse model. Here the Butterfly prediction is followed by a wavelet addition only in non-smooth regions. We have reported the corresponding distortion errors e_1 and e_2, which correspond respectively to the *MSE* (in units of 10^{-4}) and to Lavoué *et al.*'s MSDM metric [37]

7. Conclusion and Future Work

A new wavelet-based segmentation algorithm for SR meshes was introduced as a pre-processing step in a patch-independent progressive decoding of 3D meshes. Our segmentation process aims at constructing regions with different degrees ofroughness, reflected by the wavelet coefficient amplitudes (associated to each resolution level). As a preliminary step of the development of a locally-based R-D optimized coding scheme, we have studied the behavior of the wavelet decompositions in the created regions, during the patch-independent decoding and more specifically the view-dependent reconstruction process. We particularly investigated three different possible wavelet decompositions, close to the patch borders. One of them is a new consideration and has given the best results for two 3D objects remeshed by two different algorithms.

As an extension of the current work, some future research can be performed in the following areas. First, we may consider the patch-independent R-D optimized compression and coding, which could benefit from our new rule, because an independent quantization in each partition will probably allow to reduce the global distortion. New prediction schemes and coding techniques could be proposed for the non-smooth regions (where the predictions with actual subdivision schemes can be improved), together with more subjective distortion metrics, based on human vision and emphasized at the end of the previous section. Our goal will be to improve the compression bitrates on natural objects, which can be composed of non-smooth textured parts. A possible issue could consist in conceiving a statistical analysis model of the surface. The critical elements could hence serve to regenerate the same visual aspects and high frequencies of the studied surface. Another interesting consideration would be to take into account the surface anisotropy, to better predict shapes with features like sharp creases, corners or rough textures. Moreover, a fractal analysis could also be adopted in noisy regions. We could also conceive our segmentation process differently, by better analyzing the decrease of the wavelets and their relation to the surface discrete roughness or saliency.

Finally other applications could benefit from our patch-based MRA, like watermarking. It would allow to apply different marks according to the visual aspect of the surface, given the fact that a textured (or rough) region is able to hide geometric distortions much better than a smooth one.

8. Acknowledgment

I would like to address special thanks to Frédéric Payan for his R-D optimization code he gave and explained me and also for providing me the semi-regular Lion Head model remeshed by his algorithm. I don't forget Andrei Khodakovsky who has

provided me the other remeshed models. I would like to thank also Marc Antonini for the opportunity he offered me to work on his team in the I3S lab, in Sophia-Antipolis. Finally, I would like to thank the anonymous reviewers for their constructive appreciations and helpful suggestions.

References

[1] Hoppe, H. (1996). Progressive meshes, *Computer Graphics*, 30 no. Annual Conference Series, p. 99–108.

[2] Cohen-Or, D., Levin, D., Remez, O (1999). Progressive compression of arbitrary triangular meshes, *In*: *VIS'99: Proceedings of the conference on Visualization*, p. 67–72.

[3] Pajarola, R., Rossignac, J (2000). Compressed progressive meshes, *IEEE Transactions on Visualization and Computer Graphics*, 6 (1) 79–93.

[4] Valette, S., Gouaillard, A., Prost, R (2004). Compression of 3D triangular meshes with progressive precision, *Comp Graph*, 28 (1) 35–42.

[5] Khodakovsky, R., Schröder, P.,Sweldens, W (2000). Progressive geometry compression, *In*: *ACM SIGGRAPH'00*, p. 271–278.

[6] Khodakovsky, A., Guskov, I (2003). Compression of normal meshes, *Geometric Modeling for Scientific Visualization*, p. 189–206.

[7] Lavu, S.,Choi, H., Baraniuk, R (2003). Geometry compression of normal meshes using rate-distortion algorithms, *In*: *ACM SIGGRAPH'03*, 2003, p. 52–61.

[8] Payan, F., Antonini, M (2005). An efficient bit allocation for compressing normal meshes with an error-driven quantization, *Computer Aided Geometric Design - Special issue on Geometry Mesh Processing*, 22 (5) 466–486.

[9] Sim, J., Kim, C., Kuo, C., Lee, S.(2004). Rate-distortion optimized compression and view-dependent transmission of 3D normal meshes, *IEEE Transactions on Circuits and Systems for Video Technology*, 14, 854–868.

[10] Gioia, P., Aubault, O., Bouville, C (2004).Real-time reconstruction of wavelet-encoded meshes for view-dependent transmission and visualization, *IEEE Transactions on Circuits and Systems for Video Technology*, 14 (7) 1009–1020.

[11] Payan, F., Antonini, M., Mériaux, F (2009).View-dependent geometry coding of 3D scenes, *In*: *ICIP: International Conference on Image Processing*, 2009, p. 729–732.

[12] Hoppe, H. (1997). View-dependent refinement of progressive meshes, *In: SIGGRAPH '97*, 1997, p. 189–198.

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

[14] Roudet, C., Dupont, F., Baskurt, A (2007). Multiresolution mesh segmentation based on surface roughness and wavelet analysis, SPIE Visual Communications and Image Processing, 6508 (1) p. 65082E.

[15] Dyn, N., Levin, D., Gregory, J. A (1990). A butterfly subdivision scheme for surface interpolation with tension control, *ACM Trans. Graph.*, 9 (2) 160–169.

[16] Mallat, S. (1989). A theory for multiresolution signal decomposition: The wavelet representation, *IEEE Trans. Pat. Anal. Mach. Intell.*, 11, p. 674–693.

[17] Sweldens, W. (1995). The lifting scheme: A new philosophy in biorthogonal wavelet constructions, *In*: *Wavelet Applications in Signal and Image Processing III*, p. 68–79.

[18] Lounsbery, M., DeRose, T. D., Warren, J (1997). Multiresolution analysis for surfaces of arbitrary topological type, *ACM Transactions on Graphics*, 16 (1) 34–73.

[19] Eck, M., DeRose, T., Duchamp, T.Hoppe, H., Lounsbery, M., Stuetzle, W (1995). Multiresolution analysis of arbitrary meshes, *In*: *ACM SIGGRAPH'95*, p. 173–182.

[20] Gioia, P. (1999). Reducing the number of wavelet coefficients by geometric partitioning, *Computational Geometry: Theory and Applications*, 14 (1-3) 25–48.

[21] Guskov, V (2007). Manifold-based approach to semi-regular remeshing, *Graphical Models*, 69 (1). 1–18.

[22] Kammoun, A., Payan, F., Antonini, M (2010). Adaptive semi-regular remeshing: A voronoi-based approach, *In*: *Proc. of IEEE international workshop on MultiMedia Signal Processing*.

[23] Lee, A. W. F., Sweldens, W., Schröder, P., Cowsar, L., Dobkin, D (1998). MAPS: multiresolution adaptive parameterization of surfaces, *In*: *ACM SIGGRAPH'98*, V. 32, p. 95–104.

[24] Guskov, I., Vidimce, K., Sweldens, W. Schröder, P (2000), Normal meshes, *In*: *ACM SIGGRAPH'00*, x, p. 95–102.

[25] Friedel, I., Schröder,P., Khodakovsky, A (2004). Variational normal meshes, *ACM Transactions on Graphics* 23 (4) 1061–1073.

[26]Zorin, D., Schröder, P., Sweldens, W (1996). Interpolating subdivision for meshes with arbitrary topology, *In*: *ACM SIGGRAPH'96*, 1996, p. 189–192.

[27] Sim, J., Kim, C., Kuo, C., Lee, S.(2002). Normal mesh compression based on rate-distortion optimization, *In*: *IEEE Workshop on Multimedia Signal Processing*, p. 13–16.

[28] Lavoué, G., F., Dupont, F., Baskurt, A (2005). A new cad mesh segmentation method, based on curvature tensor analysis, *Computer-Aided Design*, 37 (10) p. 975–987.

[29] Gersho, A., Gray, R.M (1991). *Vector quantization and signal compression*. Kluwer Academic Publishers.

[30] Cohen-Steiner, D., Morvan, J.-M.(2003).Restricted delaunay triangulations and normal cycle, *In*: *SCG '03: Proceedings of the nineteenth annual symposium on Computational geometry*. ACM Press, 2003, p. 312–321.

[31] Lee, C. H., Jacobs, A. V., and D. W (2005). Mesh saliency, in *ACM SIGGRAPH'05*, p. 659–666.

[32] Touma,C., Gotsman, C (1998).Triangle mesh compression. *In*: *Graphics Interface*, p. 26–34.

[33] Payan, F., Antonini, M (2006). Mean square error approximation for wavelet-based semiregular mesh compression," *Transactions on Visualization and Computer Graphics (TVCG)*, V. 12.

[34] http://www.cgal.org

[36] Aspert, N., Santa-Cruz, D.,Ebrahimi, T (2002). MESH: Measuring errors between surfaces using the hausdorff distance," in *IEEE International Conference on Multimedia and Expo*, I, p. 705 – 708.

[37] Lavoué, G.., Gelasca, E. D., Dupont, F., Baskurt, A., Ebrahimi, T (2006). Perceptually driven 3D distance metrics with application to watermarking," in *SPIE Applications of Digital Image Processing XXIX*.

[38] Benhabiles, H.,Vandeborre, J.-P. Lavoué, G., Daoudi, M (2010). A comparative study of existing metrics for 3-D mesh segmentation evaluation, *The Visual Computer – International Journal of Computer Graphics*, 26 (12) 1451–1466.

Author biography

Céline Roudet received her Engineering Degree in Computer Science in 2004 and her M.S. degree in Image Processing in 2005 from INSA of Lyon (France). She received the Ph.D. degree in Computer Science with a dissertation on the surface compression with geometric wavelets, from the University of Lyon 1, France, in 2008. She is currently an associate professor at the Burgundy University in Dijon and belongs to the Le2i lab. Her research interests include 3D digital geometry processing, geometric modeling and 3D reconstruction. She is more precisely interested in 3D acquisition with active stereo vision systems and in subdivision, segmentation, remeshing and progressive compression of 3D meshes.