

# 3D MESH CODING WITH PREDEFINED REGION-OF-INTEREST

Jonas El Sayeh Khalil\*

Adrian Munteanu<sup>‡</sup>

Peter Lambert\*

\* Ghent University-IMEC, ELIS Department, Sint-Pietersnieuwstraat 41, B-9000 Ghent, Belgium

<sup>‡</sup> Vrije Universiteit Brussel, Department of Electronics and Informatics,  
Pleinlaan 2, B-1050 Brussels, Belgium

## ABSTRACT

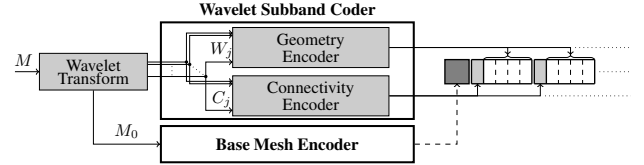
We introduce a novel functionality for wavelet-based irregular mesh codecs which allows for prioritizing at the encoding side a region-of-interest (ROI) over a background (BG), and for transmitting the encoded data such that the quality in these regions increases first. This is made possible by appropriately scaling wavelet coefficients. To improve the decoded geometry in the BG, we propose an ROI-aware inverse wavelet transform which only upscales the connectivity in the required regions. Results show clear bitrate and vertex savings. For a trivial *front-back* selection of the ROI and BG, rendering from the front saves up to 5 bits per vertex and up to 50% of the geometry, while appearing visually lossless.

**Index Terms**— Region-of-Interest, irregular mesh coding, wavelet-based coding, spatially-adaptive reconstruction

## 1. INTRODUCTION

Triangle meshes are the main representation for 3D surfaces in real-time rendering. However, real-time rendering applications have to meet specific constraints in terms of available bandwidth, memory and computational power, ultimately limiting the accuracy of the surface representation. Progressive and wavelet-based mesh representations [1] have been proposed to meet such constraints by offering scalability: in this case, the data stream can be truncated, allowing for an optimal approximation of the input mesh given a specific bitrate or computational budget.

A valuable functionality of any coding system is to allow for prioritizing specific spatial regions, e.g., prioritizing the front side of a model or the face of a character. Such functionality is called *Region-of-Interest* (ROI) coding. Vertex-by-vertex-based progressive mesh coding systems such as [2, 3] implicitly allow for prioritizing every single vertex within a mesh; hence, providing ROI coding support is trivial for such systems. However, wavelet-based mesh coding systems, e.g., [4, 5, 6, 7, 8], incrementally refine the resolution over the entire mesh. Such codecs do not provide direct access to individual vertices, hence, providing ROI support in wavelet-based coding of meshes calls for specific codec designs.



**Fig. 1.** Basic architecture of a wavelet-based mesh encoding system. A mesh  $M$  is transformed into a base mesh  $M_0$ , and a set of wavelet subbands  $W_j$  with connectivity information  $C_j$  required to reconstruct intermediate meshes  $M_j$ .

ROI support in wavelet-based coding systems has been explored in the past in the context of image coding, for instance by the JPEG-2000 image coding standard [9]. For meshes, ROI coding support has been explored for *semi-regular* mesh codecs in [10, 11]. In this paper we introduce a novel functionality providing encoder-side ROI support for *irregular* meshes; this functionality has not been proposed in the literature so far. We additionally propose an ROI-aware inverse wavelet transform to limit geometric aberrations. Such ROI capabilities allow for optimizing data transmission in order to reduce bitrate, transmission time and memory use while improving visual quality in the region(s) of interest.

The paper is structured as follows. Section 2 shortly discusses wavelet-based mesh codecs. Section 3 details our approach to enable the transmission of a predefined ROI. Section 4 discusses how we further improve the results by providing an ROI-aware inverse wavelet transform. In Section 5 we evaluate our codec and Section 6 concludes this work.

## 2. WAVELET-BASED MESH CODING

Figure 1 depicts the basic architecture for wavelet-based mesh coding systems. The *Wavelet Transform* iteratively down-samples an original mesh  $M$ , generating the multi-resolution representation  $(M=)M_{r-1}, M_{r-2}, \dots, M_0$ , with  $r$  the number of resolutions. Downsampling  $M_{j+1}$  results in a lower-resolution mesh  $M_j$ , a wavelet subband  $W_j$  and, for an *irregular* mesh codec, connectivity information  $C_j$ . The connec-

tivity of a *semi-regular* mesh is determined by a subdivision scheme, so no connectivity information is required for a semi-regular mesh codec.

The base mesh is encoded using any arbitrary single-rate coder, such as the state-of-the-art codec of Touma and Gotsman [12]. The wavelet subbands and connectivity information are encoded using the *Geometry Encoder* and *Connectivity Encoder* respectively. As indicated in Figure 1, encoding the connectivity information results in a single data block per resolution. The wavelet coefficients however are typically quantized using *Successive Approximation Quantization* (SAQ)[13] and are encoded in bitplane-by-bitplane fashion. This is a conventional approach, e.g. [7], which allows for quality scalability and rate-distortion optimization [14].

To investigate the ROI coding functionality, we employ our previously proposed wavelet-based irregular mesh codec of [8], which yields state-of-the-art compression performance for irregular meshes. In this context, we demonstrate the encoding of a predefined ROI and tackle quality degradation in the background caused by prediction error accumulation.

### 3. PREDEFINED REGION-OF-INTEREST

To encode a predefined ROI, we were inspired by the ROI coding methods defined in the JPEG-2000 image coding standard [15]. The standard defines (1) a general scaling based method which allows for scaling rectangular or elliptical regions at arbitrary scaling values, and (2) the maximum shift (maxshift) method which allows for arbitrarily-shaped regions to be encoded. The former allows for choosing for each region a relative importance w.r.t. the background (BG) but requires transmitting the ROI masks. The latter only allows for one shift, i.e., the ROI gets full precedence over the BG, but the ROI masks are arbitrarily shaped and are not encoded.

#### 3.1. Propagating an ROI mask

ROIs can be defined at every resolution. However, an  $ROI_j$  defined at a specific resolution  $j$  requires that the corresponding downscaled region  $ROI_{j \rightarrow j-1}$  is contained within  $ROI_{j-1}$ . This is described in [9] for image coding; analogue reasoning can be applied for 3D meshes.

A mask for the spatial-domain region-of-interest  $ROI_r^S$  of the original mesh  $M$  is defined by marking the vertices within the ROI. To define, for a given  $ROI_j^S$  of a higher-resolution mesh  $M_j$ , the spatial-domain  $ROI_{j-1}^S$  of the lower-resolution mesh  $M_{j-1}$ , and wavelet-domain  $ROI_{j-1}^W$  of the wavelet subband  $W_{j-1}$ , we have to look at the inverse wavelet transform. New vertices after upsampling are reconstructed using:

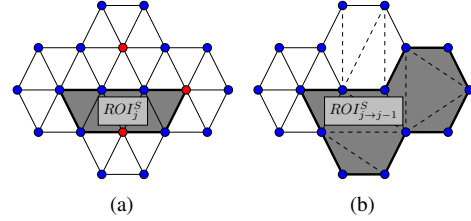
$$v_o = \text{pred}(N(v_o)) + W(v_o)$$

with  $v_o$  a new (odd) vertex in mesh  $M_j$ ,  $N(v_o)$  the set of its neighbours,  $\text{pred}(N(v_o))$  its prediction, and  $W(v_o)$  the

corresponding wavelet coefficient. Additionally, let  $v_e$  denote an even vertex in mesh  $M_j$ . It is clear that:

$$\begin{aligned} \forall v_o \in ROI_j^S : N(v_o) \subset ROI_{j-1}^S \wedge W(v_o) \in ROI_{j-1}^W \\ \forall v_e \in ROI_j^S : v_e \in ROI_{j-1}^S \end{aligned}$$

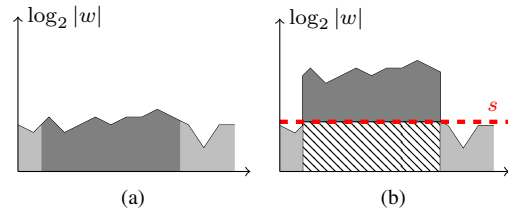
This is illustrated in Figure 2. Following this process, we obtain a base mesh  $M_0$  and a set of wavelet subbands  $W_j$  for which we have their  $ROI_j^W$ . Each  $ROI_{j-1}^S$  contains at least  $ROI_{j \rightarrow j-1}^S$ , obtained by propagating the higher-resolution  $ROI_j^S$ , but can be chosen larger if an encoder decides to prioritize other regions *only up to resolution*  $j - 1$ .



**Fig. 2.** Propagating an ROI mask.  $ROI_j^S$  on the left has the corresponding  $ROI_{j \rightarrow j-1}^S \subset ROI_{j-1}^S$  indicated on the right; the additional vertices are required for reconstructing  $ROI_j^S$ . Odd and even vertices are shown in red and blue, respectively.

#### 3.2. Boosted wavelet coefficients

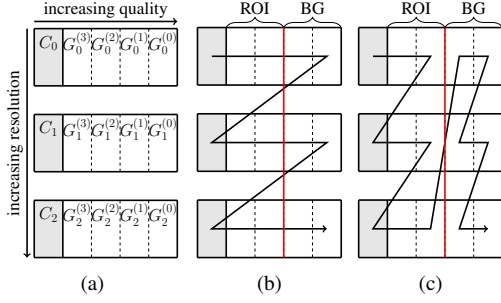
As our wavelet coefficients are quantized, we can apply the same ideas underlying the maxshift coding method: the wavelet coefficients within the ROI are upscaled by a scaling factor surpassing the largest wavelet coefficient magnitude found within the BG, as illustrated in Figure 3.



**Fig. 3.** Scaling wavelet coefficients. An example magnitude profile of wavelet coefficients  $w$  is given in Fig.3(a), with an indication of the coefficients which are within the ROI (darker gray) and the coefficients in the BG (lighter gray). Fig.3(b) shows the upscaled wavelet coefficients.

As the wavelet coefficients pertaining to the ROI occupy bitplanes with a higher significance after scaling, these wavelet coefficients will be encoded prior to the coefficients in the BG. When decoding, the decoder classifies wavelet coefficients with a magnitude larger than  $s$  as being part of the ROI and will scale them back down by this factor  $s$ , after which the classical inverse wavelet transform is performed.

### 3.3. ROI-aware transmission



**Fig. 4.** Transmission orders. Fig.4(a) shows the data layers after coding; for resolution  $j$ ,  $C_j$  represents connectivity information and  $G_j^{(i)}$  geometry information for bitplane  $i$ . Fig.4(b) shows the default resolution-scalable encoding order of [8]. Fig.4(c) depicts the ROI-aware transmission order.

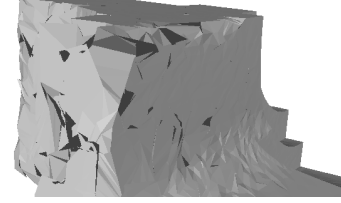
Figure 4(a) shows the data produced by the considered mesh codec [8], when using 2 bit quantization and scaling the obtained wavelet coefficients with  $s = 2^2$ . Note that this scaling value (and consequently the number of bitplanes used for the BG information) varies, in general, across subbands, depending on the largest magnitude coefficient in the corresponding subband. At each resolution level  $j$ , the system produces a connectivity layer  $C_j$  and corresponding geometry layers  $G_j^{(k)}$ . The maxshift method described above implicitly encodes the ROI before the BG, *within each subband*. This is depicted in Figure 4(b), where it is shown that the ROI bitplanes are coded before the BG bitplanes. This prioritizes the ROI information only within each resolution level, but not across resolutions: an ROI at a given resolution is only streamed after streaming both the ROI and BG information of the previous resolution. Analogue to how the rate allocator in JPEG-2000 reorders the bitstream parts across coding blocks and wavelet subbands to have the ROI streamed prior to the BG information, we need to reorder bitstream parts across resolutions.

In [14] we have shown that an arbitrary transmission order of wavelet coefficient bitplanes is possible without a negative impact on the lossless coding rate. Hence, we can make our transmission ROI-aware by streaming per resolution level the bitplanes pertaining to the ROI, subsequently followed by streaming the BG information of every resolution. This transmission order is shown in Figure 4(c).

## 4. ROI-AWARE INVERSE WAVELET TRANSFORM

Consider the transmission order depicted in Figure 4(c) and let  $q$  denote the amount of quantization bits,  $s_j$  the scaling value for subband  $j$ , and  $k_j = \log_2 s_j$ ; the decoder reads base mesh  $M_0$  and then receives  $C_0, G_0^{(q+k_0-1)}, \dots, G_0^{(k_0+1)}$ ,

$G_0^{(k_0)}$  to reconstruct  $M_1$  for which the ROI will be accurately reconstructed. Subsequently,  $C_1, G_1^{(q+k_1-1)}, \dots, G_1^{(k_1+1)}, G_1^{(k_1)}$  are received to reconstruct  $M_2$ , and so on. The ROI is again accurately reconstructed; in the BG, however, geometric errors accumulate. Figure 5 shows the back of *fandisk* after decoding the ROIs for all resolutions, i.e., up to  $G_{r-1}^{(k_{r-1})}$ .



**Fig. 5.** Geometric degradation in the BG

To obtain a smoother BG, two approaches can be taken. Either a smoothing filter is applied specifically to the BG vertices after every inverse transform step, or the inverse transform should be limited to the ROI. As we allow the connectivity to be irregular, the latter option is the most promising one as it additionally allows for reducing the amount of vertices.

### 4.1. ROI-aware inverse wavelet transform

As described in [8], performing an inverse wavelet transform on the mesh  $M_{j-1}$  entails locating patches using  $C_{j-1}$ , adding a new vertex per patch, correcting its location using the corresponding wavelet coefficient from  $W_{j-1}$  and retriangulating *within* each patch. As the border edges remain unaltered, upscaling a single patch without upscaling the other patches does not introduce topological errors. Consequently, the inverse transform can be limited to wavelet coefficients  $w \in ROI_{j-1}^W$  to accurately reconstruct  $ROI_j^S$ .

Subsequent reconstruction steps do not introduce errors either. All wavelet coefficients  $w \in ROI_j^W$  are related to odd vertices  $v_o \in ROI_{j+1}^S$ , and as Section 3.1 described,  $\forall v_o \in ROI_{j+1}^S : N(v_o) \subset ROI_j^S$ . As all neighbours of  $v_o$  are available, the patch can be retriangulated.

### 4.2. Undetected ROI vertices

The method described above is valid if all ROI masks are known. However, using the maxshift method, the decoder only finds the ROI where wavelet coefficients are non-zero. Zero-magnitude coefficients within the ROI will be determined as BG, and the corresponding patch will not be retriangulated. Subsequent ROI-aware reconstruction steps possibly depend on undetected ROI vertices, so the codec needs to store the relationships between patches and odd vertices at all resolutions, to be able to provide lower-resolution patch information (with corresponding zero-magnitude wavelet coefficients) required to reconstruct specific vertices at higher resolutions.

## 5. EVALUATION

We have compared ROI-aware encoding with the ROI-agnostic encoding of [8] using a set of 25 conventional models ranging from 1,000 up to 350,000 vertices. The wavelet coefficients were quantized using 12 bits, and we determined a scaling value per wavelet subband as illustrated in Figure 3. The ROI was defined based on the surface orientation: regions where the surface normal has a component in a specific direction are part of the ROI, i.e., the *front-facing regions*; the other regions are part of the BG, i.e., the *back-facing regions*.

Model	rate	p	s	g	vert%
teapot (1 292)	32.7	4.05	6.71	2.66	74%
drill bit (1 961)	33.3	3.29	7.01	3.72	77%
beethoven (2 521)	34.1	3.13	5.06	1.93	83%
triceratops (2 832)	32.4	2.73	4.15	1.42	87%
elk (5 194)	30.5	3.03	5.97	2.94	76%
parthenon (5 936)	27.2	2.73	5.09	2.36	74%
atomium (6 150)	26.8	2.42	4.24	1.82	80%
fandisk (6 475)	26.2	2.71	6.43	3.72	63%
maxplanck (7 399)	29.6	2.87	7.89	5.02	66%
venushead (8 268)	29.6	3.03	6.86	3.84	70%
bimba (8 857)	30.1	2.67	5.84	3.16	75%
horse (19 851)	25.1	2.59	6.23	3.64	65%
bunny (34 834)	24.2	2.26	5.61	3.35	67%
vaselion (38 728)	27.4	2.27	5.26	2.99	75%
screwdriver (65 538)	20.6	2.15	5.44	3.29	56%
rabbit (67 039)	22.6	1.60	6.07	4.47	60%
golfball (122 882)	21.8	1.24	6.26	5.02	56%
dino (129 026)	19.9	1.90	4.01	2.12	66%
headus (131 074)	19.8	1.44	4.53	3.09	58%
armadillo (172 974)	20.8	1.73	5.08	3.35	62%
igea (198 658)	19.2	1.50	5.18	3.67	52%
fertility (241 607)	21.3	1.90	6.12	4.23	57%
feline (258 046)	18.7	1.85	4.27	2.43	54%
heptoroid (286 678)	20.1	2.22	5.30	3.08	52%
skeleton hand (327 323)	19.8	2.09	5.44	3.35	50%
<b>Average</b>	<b>25.4</b>	<b>2.376</b>	<b>5.602</b>	<b>3.226</b>	<b>66%</b>

**Table 1.** Rate penalty and savings. The columns give, respectively, the models with their amount of vertices, the original rate, the rate penalty  $p$  in lossless coding in bpv, the saved rate  $s$  in bpv when viewing from the front, the final gain  $g$  in bpv, and the percentage of vertices used for this viewing angle.

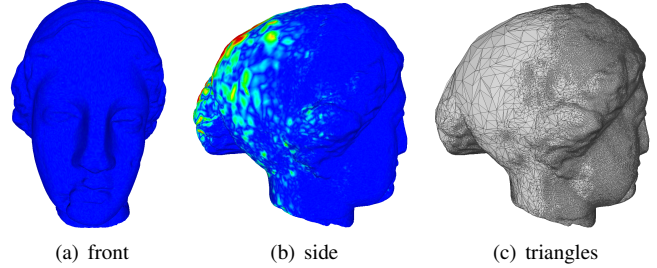
### 5.1. Penalty in lossless coding

The shifting method introduces a penalty for lossless coding due to the additional bitplanes which need to be encoded.

The second column in Table 1 gives the lossless rate for resolution-scalable coding in bits per vertex (bpv). We observe on average a coding rate of 25.4bpv, a rate which decreases with increasing number of vertices. The rate penalty introduced by allowing ROI-aware decoding is given in the third column. We observe an average rate penalty of 2.38bpv.

### 5.2. Rate decrease for ROI decoding

When viewing the models from the front, the described front-back ROI suffices to attain *visually* lossless results when only decoding the ROI. Table 1 shows an average rate saving of



**Fig. 6.** Visual results of predefined ROI-decoding for model *igea*. Blue colors represent accurate geometry reconstructions while red colors represent the largest distortions.

5.60bpv in this case. Taking into account the rate penalty, gains up to 5bpv are observed; on average 3.23bpv was saved. Furthermore, the ROI-aware inverse wavelet transform reduced the amount of vertices for such visually lossless front-view rendering by 34%. Observe also that our ROI-aware inverse transform performs better on higher-resolution data: with approximately half of the vertices in the ROI, the decoded data saves up to 50% with increasing model size.

A decreased quality can only be observed when viewing from an angle. Figure 6 shows the *igea* model, with Figure 6(a) showing the visually lossless reconstruction of the front-facing regions, and Figure 6(b) showing geometric errors in back-facing regions. However, the errors are small due to the lack of high frequency details. Figure 6(c) clearly demonstrates the adaptive inverse wavelet transform, where the front-facing regions have a high resolution, while back-facing regions are smoothed out by providing fewer vertices.

## 6. CONCLUSIONS

We have introduced predefined region-of-interest (ROI) coding to wavelet-based irregular mesh codecs. We have shown how this can be done in general by scaling wavelet coefficients, similar to the maxshift method of JPEG-2000, and how an ROI-aware inverse wavelet transform can reduce geometric errors and the required amount of vertices. A predefined ROI allows the encoder to prioritize regions if bandwidth or memory is limited. Furthermore, if the ROI is appropriately selected, lossless visual quality can be obtained at a reduced bitrate and using fewer vertices.

## 7. ACKNOWLEDGMENTS

The research activities as described in this paper were funded by Ghent University, IMEC, Flanders Innovation & Entrepreneurship (VLAIO), the Fund for Scientific Research-Flanders (FWO-Flanders), and the European Union.

## 8. REFERENCES

- [1] Faxin Yu, Hao Luo, Zheming Lu, and Pinghui Wang, *3D Mesh Compression*, chapter 2, pp. 91–160, Advanced Topics in Science and Technology in China. Springer Berlin Heidelberg, 2010.
- [2] Hugues Hoppe, “View-dependent refinement of progressive meshes,” in *Proceedings of the 24th Annual Conference on Computer Graphics - SIGGRAPH 1997*, Los Angeles, CA, US, 3–8 Aug. 1997, ACM SIGGRAPH, pp. 189–198.
- [3] Sébastien Valette, Raphaëlle Chaine, and Rémi Prost, “Progressive lossless mesh compression via incremental parametric refinement,” in *Computer Graphics Forum - Proceedings of the Eurographics Symposium on Geometry Processing - SGP 2009*, Berlin, DE, 15–17 July 2009, Eurographics, vol. 28, pp. 1301–1310, Eurographics Association Aire-la-Ville, Switzerland, Switzerland 2009.
- [4] Michael Lounsbery, Tony D. DeRose, and Joe D. Warren, “Multiresolution analysis for surfaces of arbitrary topological type,” *ACM Transactions on Graphics (TOG)*, vol. 16, no. 1, pp. 34–73, Jan. 1997, Originally available as TR-93-10-05, October, 1993, Department of Computer Science and Engineering, University of Washington.
- [5] Andrei Khodakovsky, Peter Schröder, and Wim Sweldens, “Progressive geometry compression,” in *Proceedings of the 27th Annual Conference on Computer Graphics - SIGGRAPH 2000*, New Orleans, LA, US, 23–28 July 2000, ACM SIGGRAPH, pp. 271–278.
- [6] Sébastien Valette and Rémi Prost, “Wavelet-based progressive compression scheme for triangle meshes: Wavemesh,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 10, no. 2, pp. 123–129, Mar./Apr. 2004.
- [7] Leon Denis, Shahid Mahmood Satti, Adrian Munteanu, Jan Cornelis, and Peter Schelkens, “Scalable intra-band and composite wavelet-based coding of semiregular meshes,” *IEEE Transactions on Multimedia*, vol. 12, no. 8, pp. 773–789, Dec. 2010.
- [8] Jonas El Sayeh Khalil, Adrian Munteanu, Leon Denis, Peter Lambert, and Rik Van de Walle, “Scalable feature-preserving irregular mesh coding,” *Computer Graphics Forum*, 2016.
- [9] Joel Askelöf, Mathias Larsson Carlander, and Charilaos Christopoulos, “Region of interest coding in JPEG 2000,” *Signal Processing: Image Communication*, vol. 17, no. 1, pp. 105 – 111, 2002.
- [10] Hongjuan Zheng, Bo Liu, and Hongbin Zhang, “Region-of-interest coding of 3D mesh based on wavelet transform,” in *International Conference on Image and Graphics, ICIG 2004*, Hong Kong, CN, Dec 2004, pp. 438–441.
- [11] Céline Roudet, Florent Dupont, and Atilla Baskurt, “Semi-Regular 3D Mesh Progressive Compression and Transmission based on an Adaptive Wavelet Decomposition,” in *Wavelet applications in industrial processing VI - Electronic Imaging Science and Technology Symposium*, San Jose, United States, Jan. 2009, pp. 724807–724807–12, vol. 7248.
- [12] Costa Touma and Craig Gotsman, “Triangle mesh compression,” in *Proceedings of the Graphics Interface 1998 Conference*, Vancouver, BC, CA, 18–20 June 1998, pp. 26–34.
- [13] Jerome M. Shapiro, “Embedded image coding using zerotrees of wavelet coefficients,” *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3445–3462, Dec. 1993.
- [14] Jonas El Sayeh Khalil, Adrian Munteanu, and Peter Lambert, “Rate-distortion optimized wavelet-based irregular mesh coding,” in *(To appear in) Proceedings of the 12th Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, VISIGRAPP 2017*, Porto, ES.
- [15] David S. Taubman and Michael W. Marcellin, *JPEG 2000: Image Compression Fundamentals, Standards and Practice*, Kluwer Academic Publishers, Norwell, MA, USA, 2001.