



HAL
open science

Remeshing and Spatio-Temporal Wavelet Filtering for 3D Animations

Frédéric Payan, Aymen Kammoun, Marc Antonini

► **To cite this version:**

Frédéric Payan, Aymen Kammoun, Marc Antonini. Remeshing and Spatio-Temporal Wavelet Filtering for 3D Animations. IEEE ICASSP (International Conference on Acoustics, Speech, and Signal Processing), Apr 2008, LAS VEGAS, United States. pp.IMDSP-P6.F6. <hal-00264534>

HAL Id: hal-00264534

<https://hal.science/hal-00264534v1>

Submitted on 17 Mar 2008

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

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



HAL Authorization

REMESHING AND SPATIO-TEMPORAL WAVELET FILTERING FOR 3D ANIMATIONS

Frédéric Payan, Aymen Kamoun, Marc Antonini, member, IEEE.

Laboratoire I3S (UMR 6070 CNRS-Université de Nice - Sophia Antipolis)

2000, Route des Lucioles - 06903 Sophia Antipolis - France

phone: +33 4 92 94 27 35, fax: +33 4 92 94 28 98, email: {fpayan,kamoun,am}@i3s.unice.fr

ABSTRACT

In this paper, we present a new framework to analyse and process 3D animations (defined by sequences of triangular meshes sharing the same connectivity at any frame). Our idea is to develop a spatio-temporal wavelet filtering for such data, leading to a relevant multiresolution decomposition. In geometry processing, the most efficient spatial wavelets are based on a semiregular sampling. Since the 3D animations generally have an irregular sampling, one of our contribution is a remeshing technique, transforming an animation in a sequence of semiregular meshes, which presents regularity in time but also in space. This new sampling has the advantage to improve the quality of the multiresolution decomposition in the spatial dimension. To show the contribution of such a spatio-temporal filtering in animation processing, we present some experimental results in data compression.

Index Terms— Animation, wavelet transforms, signal sampling, signal resolution, data compression.

1. INTRODUCTION

Today, a lot of works focus on the compression of animations. One animation is most of times defined by a sequence of triangular meshes sharing the same connectivity at any frame. It means that the triangulation is the same at any frame, and so the number of vertices is fixed. This temporal regularity has been already exploited with a monodimensional wavelet filtering [1, 2] in compression algorithms. On the other hand, the spatial wavelet filtering has been rarely adressed in this domain [3, 4] because of the irregular sampling of the animations, which makes the filtering complex. However, combining these two techniques to obtain a spatio-temporal wavelet filtering could be relevant, in order to improve some applications like compression, visualization or Levels of Details (LOD), for instance.

In geometry processing, the problem of irregular sampling has been overcome by a preprocessing step, transforming the irregular meshes in semiregular ones [5, 6]. The semiregular sampling of a mesh is a great advantage in comparison with an irregular sampling. first a semiregular mesh has a relevant multiresolution structure: a semiregular mesh is defined by a

coarse base mesh and several sets of regular vertices which can be added by vertex insertion, to obtain the finest resolution level (figure 1).

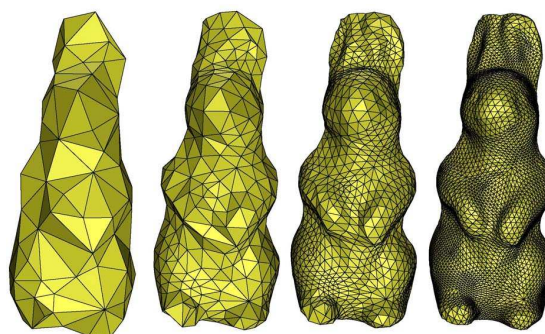


Fig. 1. Multiresolution decomposition of a semiregular mesh.

The second advantage of the semiregular meshes concerns the wavelet filtering: several works [7, 8, 9] have shown that the semiregular sampling of a mesh combined with a spatial wavelet filtering [7, 10, 11] leads to better results than coders for irregular meshes.

Since the spatial wavelet filtering is efficient for static semiregular meshes, one solution for animations should be to use a preprocessing remeshing step of the whole animation in order to ease the subsequent wavelet filtering. One work has been already proposed in this way [4], but unfortunately only the spatial dimension has been exploited with wavelets. Therefore, we propose a new framework based on wavelets to analyse and process 3D animations. Our idea is to propose a spatio-temporal wavelet filtering for such data, leading to a relevant multiresolution decomposition, which should improve some applications like compression. The first contribution of our paper is a remeshing technique for animations. The idea is obtain a semiregular representation of an animation in order to apply successively a temporal wavelet filtering and a spatial one (based on semiregular sampling). The second contribution is to show that a wavelet-based spatio-temporal decomposition of the original animation, allowing level of details in the spatial dimension but also in the temporal one, is an efficient approach to compress the animations.

The rest of this paper is organized as follows. Section 2 presents our remesher for 3D animations. Section 3 shows how creating a spatio-temporal decomposition of a remeshed animation by using wavelets. Section 4 gives some preliminary results in compression, and we conclude in Section 5.

2. PROPOSED REMESHER FOR ANIMATIONS

In order to obtain our wavelet-based spatio-temporal decomposition, we need a preprocessing stage which transforms the input animation in sequence of semiregular meshes. Our remesher is based on a motion compensation technique and the remesher MAPS [5] previously proposed for static meshes. The first step is to estimate the motion of the animation and then use this information to compensate the motion of all the frames. An *average frame* of the motion-compensated sequences is computed. Then, this *average frame* is remeshed in order to obtain a semiregular representation used as template to remesh all the frames of the animation.

2.1. Motion estimation

To compensate the motion of a given animation \mathcal{F} defined by a set of T meshes ($\mathcal{F} = \{f_1, f_2, \dots, f_t, \dots, f_T\}$), we use a classical technique of estimation/compensation. The idea is to estimate, for each frame t ($t \neq 1$), the motion of the set of vertices $\mathcal{V}_t = \{v_t^1, v_t^2, \dots, v_t^i, \dots, v_t^V\}$ in function of the first frame f_1 . The motions are defined with affine transform matrices. Then, we use the estimated motion to compensate the motion along the whole animation, and finally obtain the so-called “motion-compensated sequence”:

$$\hat{\mathcal{F}} = \{\hat{f}_1, \hat{f}_2, \dots, \hat{f}_t, \dots, \hat{f}_T\},$$

where each mesh \hat{f}_t is now defined by a new set of vertices

$$\hat{\mathcal{V}}_t = \{\hat{v}_t^1, \hat{v}_t^2, \dots, \hat{v}_t^i, \dots, \hat{v}_t^V\}.$$

Currently, we use the technique of [12], but any similar technique can be used ([13] for instance). The figure 2 shows one result obtained with this technique.

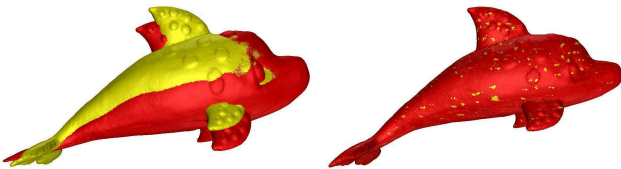


Fig. 2. Effect of a motion compensation on the DOLPHIN animation (101 frames). At left, the original frames 1 (in yellow) and 70 (in red) without compensation ; at right, the motion of the frame 70 is compensated.

2.2. Creation of the base semiregular representation

Once the motion-compensated sequence obtained, an *average frame* F_a is computed. Each vertex $v_a^j \in F_a$ is computed by: $v_a^j = \sum_{i=0}^T \hat{v}_i^j$, with \hat{v}_i^j the j^{th} vertex of the motion-compensated frame \hat{F}_i . This resulting frame is used as template to obtain the semiregular structure, created with a modified version of the remesher MAPS [5].

MAPS proposes 4 steps: i) the input irregular mesh is decimated to create a coarse mesh (see figure 1, on the left); ii) a parameterization of the decimated vertices is done on this coarse mesh; iii) this parameterization is used to add regular vertices on the coarse mesh by triangle subdivision (see figure 1, the second on the left); iv) the third step is repeated until the finest resolution level (see figure 1, on the right). For more explanations about MAPS, please refer to [5].

Our modification of the state-of-the art method lies in the decimation technique. Instead of vertex collapses, we use the *Qslim* algorithm of Garland and Heckbert [14], with half-edge collapses. Also, we include additional tests, such the topological one of [15], to ensure a manifold mesh. Finally, we obtain a semiregular version of the *average frame* F_a , which will be used as template to remesh the other frames of the sequence.

2.3. Remeshing of the whole sequence

To create a semiregular version at any frame f_t , we try to find the positions of all the vertices of the remeshed *average frame* on all the subsequent frames. There are two cases:

1. the given vertex is one irregular vertex of the coarse *average frame* (this vertex has not been removed during the decimation step). In this case, its motion has been already computed during the motion compensation step. So we can easily find its location for all the subsequent frames.
2. the given vertex is one regular vertex added by subdivision during the refinement step of the coarse *average frame*. In this case, its motion is unknown and we need to estimate it. For this, we use the parameterization domain previously computed with MAPS. This vertex belongs obviously to one triangle (v_m, v_n, v_o) of the parametric domain (on the original *average frame*). Thus its location is given by $v = \alpha v_m + \beta v_n + \gamma v_o$, with (α, β, γ) its barycentric coordinates. Finally, the motion M of this vertex is obtained by interpolating the known motions of the three vertices of the concerned triangle: $M = \alpha M_m + \beta M_n + \gamma M_o$, where M_m, M_n , and M_o are respectively the motion (affine transforms matrix) of the vertices v_m, v_n and v_o .

By applying the same technique for all the frames, the semiregular version of the whole animation is obtained. The figure 3 shows the result on the frame 51 of the DOLPHIN animation.

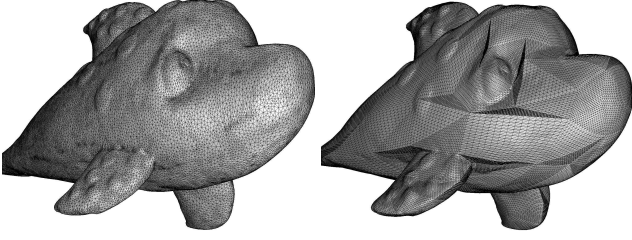


Fig. 3. Input and output of the proposed remesher: the DOLPHIN animation at the frame 51. The irregular version (on the left) has 47773 vertices and 95542 triangles, and the semiregular one (on the right) has 51202 vertices and 102400 triangles.

3. SPATIO-TEMPORAL DECOMPOSITION

To create a spatio-temporal decomposition of this semiregular animation, we first apply the temporal wavelet filtering of [2]. This filtering is based on a lifting scheme. The principle of this analysis is to apply a monodimensional filtering on the successive temporal positions of each vertex. After this step, we obtain two subsets: a high frequency subband H_1^t defined by $T/2$ detail frames (*temporal wavelet coefficients*): $H_1^t = \{h_{1,0}^t, h_{1,1}^t, \dots, h_{1,T/2}^t\}$; a low frequency subband L_1^t defined by $T/2$ approximation frames (*temporal low frequency signal*): $L_1^t = \{l_{1,0}^t, l_{1,1}^t, \dots, l_{1,T/2}^t\}$. One can obtain a multiresolution decomposition by subsequent filterings of the low frequency subbands. This decomposition consists in N subbands of detail frames $H_{r^t}^t$ (where r^t represents the temporal resolution), and the low frequency sequence L_N^t .

Then, we apply the spatial wavelet filtering based on the butterfly lifting scheme [7] on all the frames of the temporal low frequency sequence L_N^t . For each frame $l_{N,k}^t$ of this sequence, this step gives M subbands of spatial wavelet coefficients $h^s l_{r^s,k}^t$ (where r^s represents the spatial resolution level) and a coarse version $l^s l_{M,k}^t$ of the given frame.

The principle of this decomposition is illustrated by the figure 4. On this figure, we apply 2 levels of temporal decomposition, and 2 levels of spatial decomposition.

4. ONE APPLICATION IN COMPRESSION

4.1. Compression algorithm used

To show the interest of such a spatio-temporal decomposition, we propose the compression algorithm given by the figure 5. To take into account the spatio-temporal decomposition, we combine the temporal wavelet-based coder of [2] and the zerotree coder of [7] as follow:

- we encode the temporal wavelet coefficients (H_i^t) with the temporal wavelet-based coder of [2], by imposing a global target bitrate (R_T);

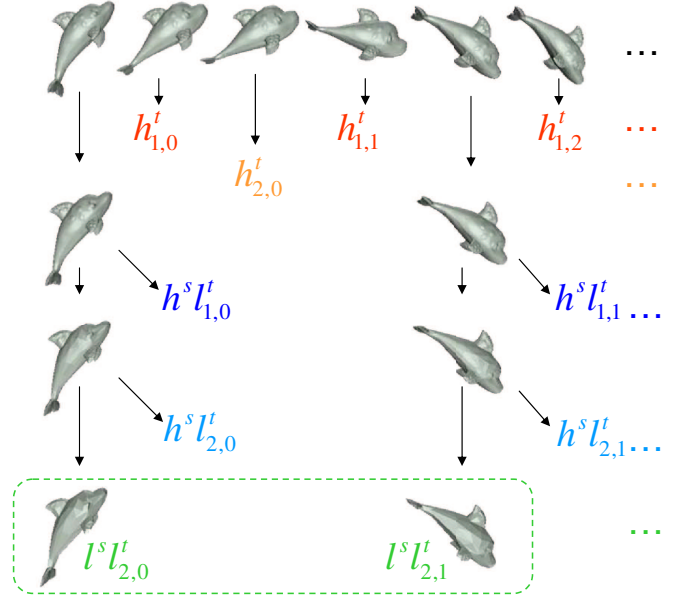


Fig. 4. Spatio-temporal decomposition based on wavelet filtering: 2 levels of temporal decomposition giving the temporal wavelet coefficients (in red and orange), and 2 levels of spatial decomposition giving the spatial ones (in blue). The meshes in the dotted rounded box (in green) represent the coarse sequence (spatio-temporal low frequency signal).

- then we encode the set of spatial wavelet coefficients ($\{H^s L_{r^s}^t\}$) with the zerotree coder of [7] by imposing the optimal bitrate (R_S) computed by the allocation process of [2] for the encoding of the temporal low frequency subband.

4.2. Results

The figure 6 shows the curve “rate-distortion” of the DOLPHIN obtained with 3 methods: i) using a temporal wavelet filtering *without* remeshing (coder of [2]); ii) using a temporal wavelet filtering *after* remeshing (proposed remesher and coder of [2]); iii) using a spatio-temporal wavelet filtering *after* remeshing (proposed remesher and proposed coder).

As distortion criterion, we use the metric based on Hausdorff distance given in [16], which is generally used to evaluate the quality of compressed meshes.

The bitrate is given in bits *per irregular vertex* and per frame.

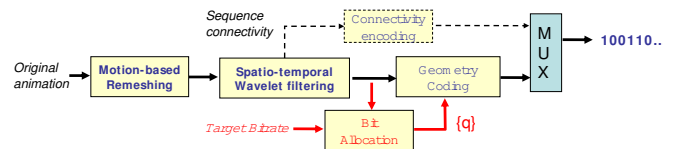


Fig. 5. Proposed compression algorithm.

We observe that the performances are significantly better when using our remesher before the compression step. Moreover, we observe that the approach based on our spatio-temporal decomposition gives the best results. It proves the interest of a spatio-temporal decomposition of an animation, and consequently the interest of a remeshing preprocessing before compression.

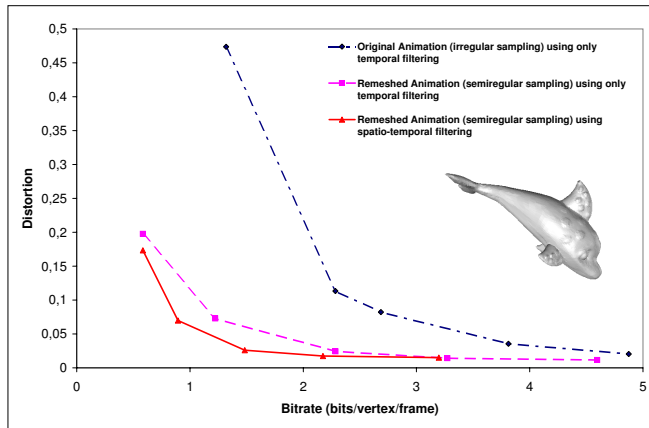


Fig. 6. Comparison of the different methods.

5. CONCLUSION AND FUTURE WORKS

We have presented in this paper a new framework to analyse and process 3D animations. We highlight the interest of using a spatio-temporal wavelet filtering on animations. We show that it allows a multiscale decomposition, in time and space, which improves significantly the compression performances. The spatial filtering being more efficient for regular samplings, we also propose a simple method of remeshing which transforms the input animation into a sequence of semiregular meshes. These contributions are preliminary works, and there are numerous ways to explore in order to improve the performances of our remesher and of the proposed compression algorithm.

6. ACKNOWLEDGEMENTS

This work was partially supported by GDR ISIS-CNRS 720, with the national project AMOA3D. We are particularly grateful to J-H. Yang for providing us his DOLPHIN, and to A. Elkefi for generating the used animation.

7. REFERENCES

- [1] A.C. Lopes and M.N. Gamito, “Wavelet compression and transmission of deformable surfaces over networks,” in *Proc. of the 10th Portuguese Computer Graphics Meeting*, 2001, pp. 107–114.
- [2] F. Payan and M. Antonini, “Temporal wavelet-based geometry coder for 3d animations,” *Computer & Graphics*, vol. 31, pp. 77–88, january 2007.
- [3] I. Guskov and A. Khodakovsky, “Wavelet compression of parametrically coherent mesh sequences,” in *Eurographics/ACM SIGGRAPH Symposium on Computer Animation*, August 2004.
- [4] JH. Yang, CS. Kim, and SU. Lee, “Semi-regular representation and progressive compression of 3D dynamic mesh sequences,” *IEEE Trans. on Image Processing*, vol. 15, no. 9, pp. 2531–2544, 2006.
- [5] A.W.F Lee, W. Sweldens, P. Schröder, P. Cowsar, and D. Dobkin, “MAPS: Multiresolution adaptive parameterization of surfaces,” *SIGGRAPH*, 1998.
- [6] I. Guskov, K. Vidimce, W. Sweldens, and P. Schröder, “Normal meshes,” in *Siggraph 2000, Computer Graphics Proceedings*, 2000.
- [7] A. Khodakovsky, P. Schröder, and W. Sweldens, “Progressive geometry compression,” *Proceedings of SIGGRAPH*, 2000.
- [8] A. Khodakovsky and I. Guskov, “Normal mesh compression,” *Geometric Modeling for Scientific Visualization*, Springer-Verlag, 2002.
- [9] F. Payan and M. Antonini, “An efficient bit allocation for compressing normal meshes with an error-driven quantization,” *Computer Aided Geometric Design*, vol. 22, pp. 466–486, July 2005.
- [10] D. Li, K. Qin, and H. Sun, “Unlifted loop subdivision wavelets,” in *Pacific Graphics Conference on Computer Graphics and Applications*, 2004, pp. 25–33.
- [11] M. Bertram, “Biorthogonal loop-subdivision wavelets,” *Computing*, vol. 72, no. 1-2, pp. 29 – 39, 2004.
- [12] Y. Boulfani, M. Antonini, and F. Payan, “Motion-based mesh clustering for mcdwt compression of 3d animated meshes,” in *Proceedings of EUSIPCO*, 2007.
- [13] K. Mamou, T. Zaharia, and F. Preteux, “A skinning approach for dynamic 3d mesh compression,” *Computer Animation and Virtual Worlds*, July 2006.
- [14] M. Garland and PS. Heckbert, “Surface simplification using quadric error metrics,” *Proceedings ACM SIGGRAPH’97*, 1997.
- [15] T. Dey, H. Edelsbrunner, S. Guha, and D. Nekhayev, “Topology preserving edge contraction,” 1998.
- [16] N. Aspert, D. Santa-Cruz, and T. Ebrahimi, “Mesh: Measuring errors between surfaces using the hausdorff distance,” in *Proceeding of ICME*, 2002.