Adaptive Semi-Regular Remeshing: A Voronoi-Based Approach

Aymen Kammoun [1], Frédéric Payan [2], Marc Antonini [3]

Laboratory I3S, University of Nice-Sophia Antipolis / CNRS (UMR 6070) - France

[1] kammoun@i3s.unice.fr

[2] fpayan@i3s.unice.fr

[3] am@i3s.unice.fr

***Abstract*—We propose an adaptive semi-regular remeshing algorithm for surface meshes. Our algorithm uses Voronoi tessellations during both simplification and refinement stages. During simplification, the algorithm constructs a first centroidal Voronoi tessellation of the vertices of the input mesh. The sites of the Voronoi cells are the vertices of the base mesh of the semi-regular output. During refinement, the new vertices added at each resolution level by regular subdivision are considered as new Voronoi sites. We then use the Lloyd relaxation algorithm to update their position, and finally we obtain uniform semi-regular meshes. Our algorithm also enables adaptive remeshing by tuning a threshold based on the mass probability of the Voronoi sites added by subdivision. Experimentation shows that our technique produces semi-regular meshes of high quality, with significantly less triangles than state of the art techniques.**

I. I

Dense meshes created by 3D scanners or 3D modelling software programs often have an irregular sampling. Because of their irregularity and size, these meshes are awkward to handle in such common tasks as storage, display or transmission. It is now established that multiresolution structures are well-suited to address these issues. Several approaches exist. The first approach is to simplify progressively the input mesh to get several levels of detail [1]. The second approach is to extend multiresolution analysis to surface meshes, by using either "irregular wavelets" [2] directly on the input mesh, or wavelets based on subdivision connectivity [3], [4], [5] on *remeshed* semi-regular meshes.

A. Semi-Regular Meshes

A semi-regular mesh M_{sr} is based on a mesh hierarchy M_{sr}^{l} ($l \in \{0, 1, ..., L\}$) that represents a given surface at different levels of resolution. M_{sr}^{0} corresponds to the lowest resolution, and is called the *base mesh*. M_{sr}^{l} is a subdivided version of M_{sr}^{l-1}, and corresponds to l^{th} level of resolution. M_{sr}^{L} corresponds to the highest resolution of the given surface (see Fig. 1, top right).

We call *semi-regular remeshing* the process transforming an irregular mesh M_{ir} into a semi-regular one M_{sr}. This process generally consists in:

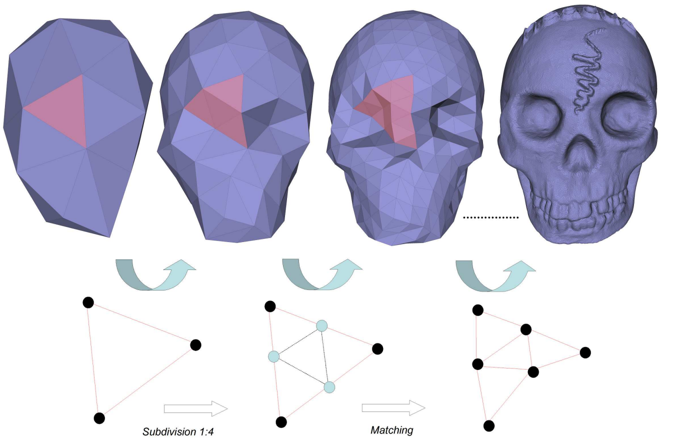


Fig. 1. One example of semi-regular meshes. Top, from left to right: the base mesh, the first, the second and the finest level of resolution of S . Bottom, the subdivision procedure that produces the different levels of resolution.

- decimating the irregular mesh M_{ir} to obtain the base mesh M_{sr}^{0}. We refer to this stage as *simplification*.
- refining M_{sr}^{0} several times to obtain the intermediary resolutions M_{sr}^{1}, M_{sr}^{2} and so on, until the final one M_{sr}^{L}. This *refinement* is based on a subdivision scheme. This consists in splitting each triangle into four smaller ones by adding new vertices on each edge, and then updating their position to fit as closely as possible to the original surface (see Fig. 1, bottom).

Semi-regular meshes have several advantages, for instance:

- the surfaces are displayed easily at different levels of details, according to the screen resolution, or the position from the camera. This makes the transmission and the displaying faster;
- the surfaces are compressed more efficiently. It is now established that wavelet-based coders taking as input semi-regular meshes are the most efficient [5], [6], [7].

B. Related Works in Semi-regular Remeshing

We first cite the work of Eck et al. [8] that consists in building a parameterization based on Voronoi tiling and harmonic maps. In 1998, Lee and al. [9] proposed a remeshing technique based on vertex collapse and conformal mapping.

MMSP'10, October 4-6, 2010, Saint-Malo, France.
978-1-4244-8112-5/10/$26.00 ©2010 IEEE.

Guskov and al. then introduced an algorithm producing semi-regular meshes such as each level of resolution l is a normal offset of the $(l+1)$ resolution level [10]. This algorithm gives good compression results [11], but works only for closed surfaces. Kyu-Yeul and al. improved this method for surfaces with boundaries in 2002 [12].

In 2007, Guskov presented another semi-regular remeshing based on global parameterization that is smooth with respect to a differential structure built on the base domain [13]. This method also exploits a vertex-based Voronoi tessellation to construct the base mesh.

C. *Proposed Approach and Contributions*

We propose an original semi-regular remeshing based on Voronoi tessellation. The Voronoi tessellation is well known in geometry processing [14]. In (re)meshing for instance, Valette *et al.* developed a resampling technique for surface meshes based on a metric-driven discrete Voronoi tessellation construction [15]. Alliez *et al.* also proposed a variational surface reconstruction based on Voronoi tessellation [16].

The originality of our method is that Voronoi tessellation are used not only during simplification [13] but also during refinement to fit the vertices on the original surface.

a) Simplification: a first centroidal Voronoi tessellation of the geometry of the irregular mesh M_{ir} is computed. The sites of the Voronoi cells are the vertices of the base mesh.

b) Refinement: the first contribution is that we consider the new vertices of M^1_{sr} added by subdivision of M^0_{sr} as *additional* Voronoi sites. Then, the Lloyd relaxation algorithm is applied again to optimize their position. By applying successively this procedure on the resulting mesh, we finally obtain semi-regular meshes uniformly sampled, and visually close to the original surface (represented by M_{ir}).
Our second contribution is a criterion based on the density of the geometry of the input mesh, that allows us to refine only triangles located in highly detailed regions. This technique produces adaptive semi-regular meshes.

The rest of this paper is organized as follows. Section II briefly introduces the Voronoi tessellation. Section III presents the two steps of our algorithm, simplification and refinement. Section IV shows some experimental results. We finally conclude in section V.

II. V T

A. *Definition in the continuous case*

Let us introduce an open set $\mathscr{S} \subset \mathbb{R}^n$, and K different sites $\{s_k;\ k = \{0, 1, ..., K-1\}\}$. The Voronoi tessellation (or *diagram*) can be defined as the set of K distinct *cells* (or *regions*) R_k such that:

$$R_k = \{v \in \mathscr{S} \mid d(v, s_k) \leq d(v, s_j);\ j \neq k\}, \tag{1}$$

with d a specific distance. The dual of a Voronoi tessellation is a Delaunay triangulation.

B. *Centroidal Voronoi Tessellation*

A *centroidal* Voronoi tessellation is a Voronoi tessellation where each site s_k is also the mass centroid of the Voronoi cell R_k [17]:

$$s_k = \frac{\int_{R_k} v\rho(v)dv}{\int_{R_k} \rho(v)dv}, \tag{2}$$

where $\rho(v)$ is a density function. Centroidal Voronoi tessellations minimize the following energy:

$$E = \sum_{k=0}^{K-1} \int_{R_k} \rho(v)||v - s_k||^2 dv. \tag{3}$$

Those tessellations are widely used in applications that require a good sampling of the input domain. One way to distribute a set of points isotropically and in accordance with a density function is to apply the Lloyd relaxation [18] over an initial tessellation. The Lloyd relaxation consists in iterating the following stages:

- each region R_k of the Voronoi tessellation is integrated and the site s_k is computed by using Eq. (2);
- each point v_i is then associated to the Voronoi region R_k which is represented by the nearest site s_k to the point v_i by using the nearest neighbour rule of Eq. (1).

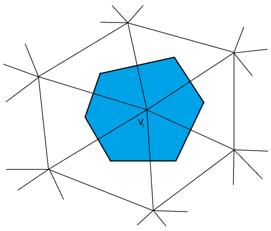


Fig. 2. Neighbourhood of a vertex v_i. The blue region is the dual cell of v_i; its area represents the density $\rho(v_i)$.

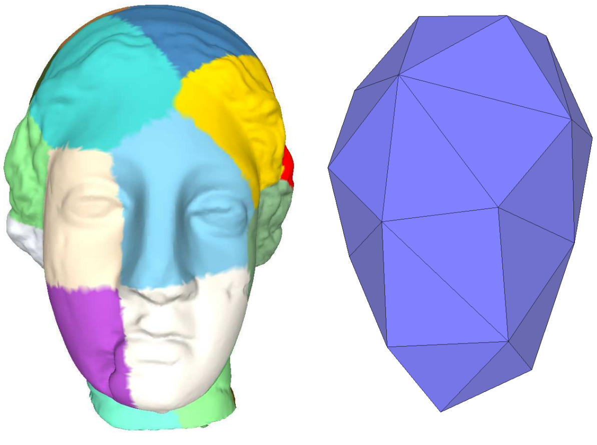
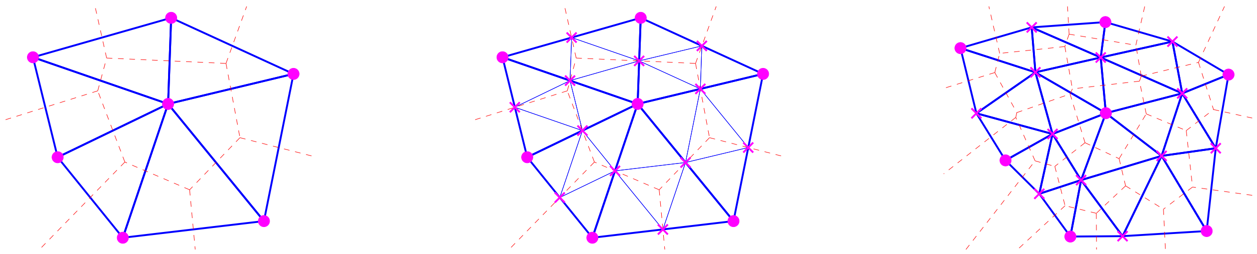


Fig. 3. Left: Voronoi tessellation of V . Right: simplified version used as base mesh for the semi-regular output M_{sr}.



(a) A patch of triangles of resolution $(l-1)$ is shown in blue. The Voronoi tessellation is shown in red.

(b) the triangles are subdivided; each new vertex, (represented by '+'), is an additional Voronoi site.

(c) Each additional Voronoi site is initialized by the nearest vertex v_i of M_{ir}, and the Voronoi tessellation is updated by Lloyd relaxation.

Fig. 4. Principle of an iteration of our refinement procedure.

C. Centroidal Voronoi Tessellation in 3D meshes

In our case, the input data are surface meshes. The set $\mathscr{S}$ is the set of vertices v_i of the irregular mesh M_{ir}. Moreover, we choose the density ρ of a given v_i of M_{ir} as the area of the dual cell of v_i as shown by Fig. 2. We choose the distance d in Eq. (1) as the Euclidean distance (in the 3D space), suited for smooth and densely sampled meshes. The site s_k of a Voronoi region R_k is the nearest vertex v_i in the original mesh M_{ir} to the mass centroid of R_k. See Fig. 3 (left) for an example of Voronoi tessellation on V .

III. P M

A. Simplification

We recall that this stage produces the base mesh M^0_{sr} from the irregular mesh M_{ir}. Let us denote s^l_k as the k^{th} site at resolution l. The goal is to create a Voronoi tessellation with few sites s^0_k. Those sites will correspond to the vertices of M^0_{sr}. Once the sites selected, all the other vertices of M_{ir} will be decimated to obtain M^0_{sr}.

We choose to use the Linde–Buzo–Gray (LBG) algorithm [19] to determine those sites. The LBG algorithm consists in creating an initial Voronoi tessellation composed by only one region, and then iterating the following process until obtaining a given number of sites:

- split the site s^0_k of each region R^0_k into two sites (the number of cells is multiplied by two);
- use the Lloyd relaxation until convergence to update the tessellation.

Once the sites selected, we choose to remove the rest of the vertices of M_{ir} by using vertex collapses, similarly to [9]. During this process, the parameterization of [9] is also constructed. We will use this parameterization during refinement.

B. Refinement

The objective is to produce the output semi-regular mesh M_{sr} from the coarse mesh M^0_{sr}. The refinement is done iteratively. For a given iteration, the procedure to obtain the mesh M^l_{sr} from M^{l-1}_{sr} is the following (see Fig. 4):

- M^{l-1}_{sr} is subdivided by associating one new vertex at each edge (see Fig. 4(b));
- we consider each new vertex as a new Voronoi site s^l_k. Its position is initialized by the nearest vertex v_i of M_{ir}, by using parameterization computed during simplification;
- the Lloyd relaxation is finally applied on the set of sites $\{s^m_k;\ m = \{0, 1, ..., l\}\}$, to obtain the semi-regular mesh M^l_{sr} (see Fig. 4(c)).

The limitation of this algorithm is that subdividing uniformly all the triangles is not always relevant. Indeed, it is useless to subdivide flat regions, contrary to highly detailed regions. Finally the resolution of each triangle should be adapted to the local geometric features of the input surface. Therefore, we propose to make this algorithm adaptive. For this purpose, we first define, at each resolution level l, a probability mass function f associated to the site s^l_k such as:

$$f(s^l_k) = |R^l_k|, \tag{4}$$

where $|R^l_k|$ is the number of vertices of M_{ir} associated to R^l_k, the k^{th} Voronoi region at the resolution l. Then, for each triangle t^l of M^l_{sr} composed by three Voronoi sites $s^l_{k_1}$, $s^l_{k_2}$ and $s^l_{k_3}$, we associate a probability mass function g defined by:

$$g(t^l) = \min_{q\in\{k_1,k_2,k_3\}} (f(s^l_q)). \tag{5}$$

Once the probabilities are computed, we choose to subdivide only the triangles with a probability higher than a user-given threshold ϵ. This procedure enables to subdivide only the highly detailed regions since there are generally more vertices in those regions. This is typically true for meshes which are man-made.

An outline of our refinement procedure is shown by Fig. 5.

IV. E R

Using a geodesic distance instead of the 3D Euclidean distance (as proposed in section II-C) seems more relevant when applying Voronoi tessellation on surfaces. But the bias induced by those two distances are low when dealing with densely sampled meshes [15]. Thus in our context, the Voronoi tessellation would be efficient if M_{ir} is dense in comparison

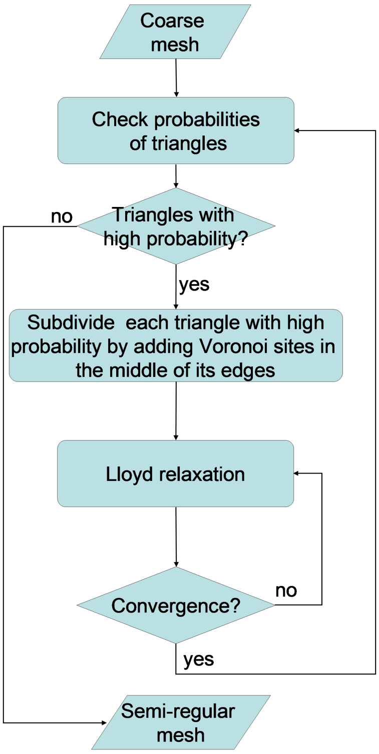

Coarse mesh
Check probabilities of triangles
Triangles with high probability?
no
yes
Subdivide each triangle with high probability by adding Voronoi sites in the middle of its edges
Lloyd relaxation
Convergence?
no
yes
Semi-regular mesh

Fig. 5. Outline of the refinement procedure.

to M_{sr}. Irregular meshes with a too small number of vertices must be oversampled before remeshing.

To know the most relevant oversampling scheme for our method, we first compare the performances of our remeshing on different models subdivided three times with the approximating Loop [20] or with the interpolating Butterfly [21] schemes. Fig. 6 shows the evolution of the remeshing error during the refinement procedure, each dot corresponding to an iteration. The remeshing error is the Root Mean Square Error stimulated by the Hausdorff distance presented in [22]. This allows us to measure a mean distance between two surfaces with different samplings. We observe that the object is better remeshed when the input data is oversampled with the Butterfly filter (lower remeshing error). Similar results are obtained on other objects. Therefore, for the subsequent experimentation, input data are oversampled with Butterfly before remeshing.

In order to observe the impact of the threshold ϵ during refinement, we have also remeshed V with different values of ϵ. Fig. 7 shows the evolution of the remeshing error. We observe that the remeshing error decreases faster when using high thresholds. This observation was expected since the remeshing error essentially decreases when subdividing the highly detailed regions.

Now we compare our results with the remeshers of [9] and [10]. Fig. 7, 8 and 9 show the evolution of the remeshing error V , S and R (for S , (we did not find S remeshed with [9]). We observe that for a given remeshing error, our approach strongly reduces the number of triangles of the semi-regular output (compared to [9] and [10]).

Fig. 10 shows some visual results for S . We clearly observe that we obtain a higher visual quality with our method, while reducing the number of triangles in the smooth regions. We also observe that there are much more triangles in highly detailed regions (on the scar).

Finally, Fig. 11 and 12 show additional visual results for V and R , respectively.

V. C

In this paper, we proposed an original semi-regular remeshing for surface meshes. A multiresolution Voronoi tessellation is constructed upon the original mesh by getting a coarse one and then adding Voronoi sites adaptively. We obtain semi-regular meshes of higher quality with our approach, and experimental results confirm the interest of our adaptive Voronoi-based approach. When comparing with two state of the art remeshers, we observed that:

- at low resolution, where the subdivision is performed uniformly, our method obtains the smallest remeshing error (at these levels, all the triangles present high probability). This proves the interest of using a Voronoi-based approach;
- at higher resolution, where the subdivision becomes adaptive, we also obtain smallest remeshing errors while reducing the number of triangles by almost a factor 2. This proves the interest of an adaptive approach.

Future works will concern:

- the study of our algorithm on more densely and more complex meshes;
- the adaptation of our method to clouds of points, to produce semi-regular meshes directly from 3D scanners.

A

V and R are courtesy of Cyberware and S is courtesy of Headus. We are particularly grateful to Igor Guskov for providing us with his normal meshes. The MAPS version of V can be found at: http://www.multires.caltech.edu/software/pgc/.

R

[1] H. Hoppe, “Progressive meshes,” in *SIGGRAPH ’96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM, 1996, pp. 99–108.

[2] S. Valette and R. Prost, “Wavelet based multiresolution analysis of irregular surface meshes,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 10, no. 2, march/april 2004.

[3] A. Certain, J. Popovic, T. DeRose, T. Duchamp, D. Salesin, and W. Stuetzle, “Interactive multiresolution surface viewing,” in *SIGGRAPH ’96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM, 1996, pp. 91–98.

[4] M. Lounsbery, T. DeRose, and J. Warren, “Multiresolution analysis for surfaces of arbitrary topological type,” *ACM Trans. Graph.*, vol. 16, no. 1, pp. 34–73, 1997.

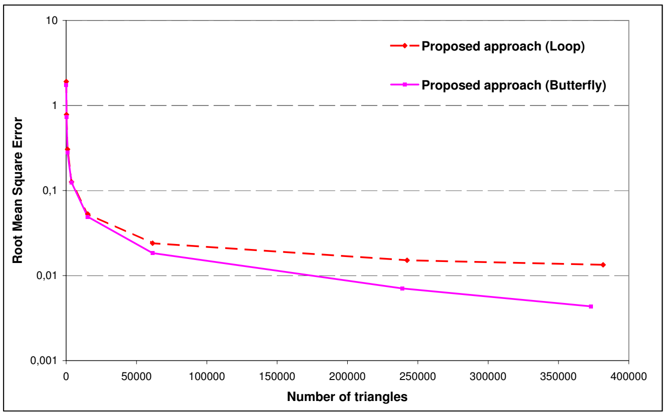


Fig. 6. Evolution of the remeshing error (logarithmic scale) for V during the refinement stage. The input mesh is subdivided three times with Loop filter (dashed plot) or Butterfly filter (continuous plot).

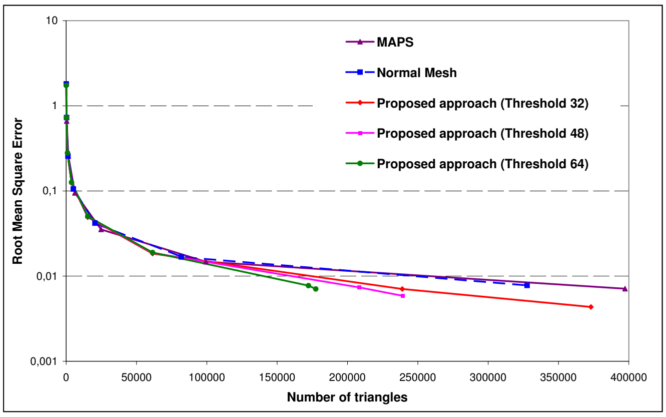


Fig. 7. Evolution of the remeshing error (logarithmic scale) for V during the refinement stage.

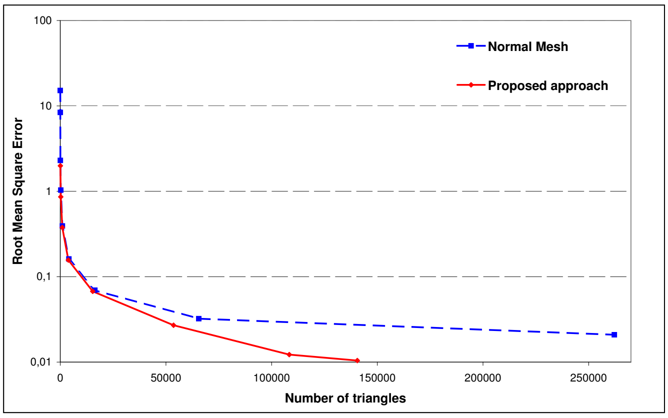


Fig. 8. Evolution of the remeshing error (logarithmic scale) for S during the refinement stage with threshold $\epsilon = 32$.

[5] A. Khodakovsky, P. Schröder, and W. Sweldens, "Progressive geometry compression," in *SIGGRAPH '00: Proceedings of the 27th annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., 2000, pp. 271–278.

[6] A. Khodakovsky and I. Guskov, "Compression of normal meshes," in *Geometric Modeling for Scientific Visualization*. Springer-Verlag, 2003, pp. 189–206.

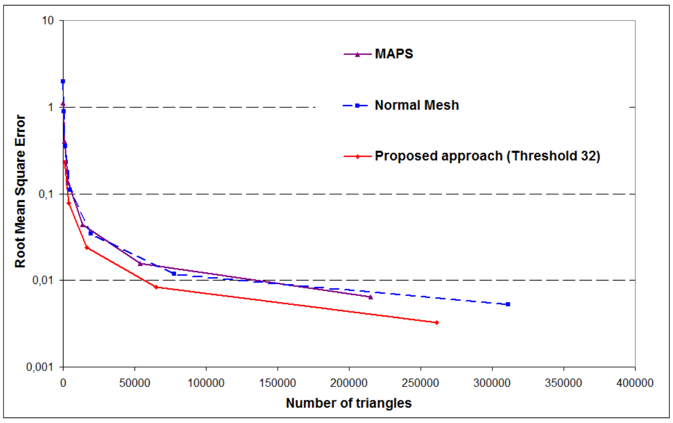


Fig. 9. Evolution of the remeshing error (logarithmic scale) for R during the refinement stage.

[7] F. Payan and M. Antonini, "An efficient bit allocation for compressing normal meshes with an error-driven quantization," *Computer Aided Geometry Design*, vol. 22, no. 5, pp. 466–486, 2005.

[8] M. Eck, T. Derose, T. Duchamp, H. Hoppe, M. Lounsbery, and W. Stuetzle, "Multiresolution analysis of arbitrary meshes," 1995.

[9] A. W. F. Lee, W. Sweldens, P. Schröder, L. Cowsar, and D. Dobkin, "MAPS: Multiresolution adaptive parameterization of surfaces," in *Proceedings of SIGGRAPH 98*, Jul. 1998, pp. 95–104.

[10] I. Guskov, K. Vidimce, W. Sweldens, and P. Schröder, "Normal meshes," in *SIGGRAPH 2000*, 2000.

[11] A. Khodakovsky and I. Guskov, "Compression of normal meshes," in *Geometric Modeling for Scientific Visualization*, Springer-Verlag, Ed., 2003.

[12] L. Kyu-Yeul, K. Seong-Chan, and K. Tae-Wan, "Normal meshes for multiresolution analysis of irregular meshes with boundaries," *JSME international journal Series C, Mechanical systems, machine elements and manufacturing*, 2002.

[13] I. Guskov, "Manifold-based approach to semi-regular remeshing," *Graph. Models*, vol. 69, no. 1, pp. 1–18, 2007.

[14] A. Franz, "Voronoi diagrams—a survey of a fundamental geometric data structure," *ACM Comput. Surv.*, vol. 23, no. 3, pp. 345–405, 1991.

[15] S. Valette, J. M. Chassery, and R. Prost, "Generic remeshing of 3d triangular meshes with metric-dependent discrete voronoi diagrams," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 2, pp. 369–381, 2008.

[16] P. Alliez, D. Cohen-Steiner, Y. Tong, and M. Desbrun, "Voronoi-based variational reconstruction of unoriented point sets," in *SGP '07: Proceedings of the fifth Eurographics symposium on Geometry processing*. Aire-la-Ville, Switzerland, Switzerland: Eurographics Association, 2007, pp. 39–48.

[17] Q. Du, V. Faber, , and M. Gunzburger, "Centroidal voronoi tessellations: applications and algorithms," in *SIAM Review, no. 41(4)*, 1998.

[18] S. P. Lloyd, "Least squares quantization in pcm," *IEEE Transactions on Information Theory*, vol. 28, pp. 129–137, 1982.

[19] Y. Linde, A. Buzo, and R. Gray, "An algorithm for vector quantizer design," *IEEE Transactions on Communications*, vol. 28, pp. 84–94, 1980.

[20] C. Loop, "Smooth subdivision surfaces based on triangles," Master's thesis, University of Utah, 1987.

[21] N. Dyn, D. Levin, and J. A. Gregory, "A butterfly subdivision scheme for surface interpolation with tension control," *ACM Transactions on Graphics*, vol. 9, pp. 160–169, 1990.

[22] N. Aspert, D. Santa-Cruz, and T. Ebrahimi, "Mesh: Measuring errors between surfaces using the hausdorff distance," in *IEEE International Conference in Multimedia and Expo (ICME)*, 2002.

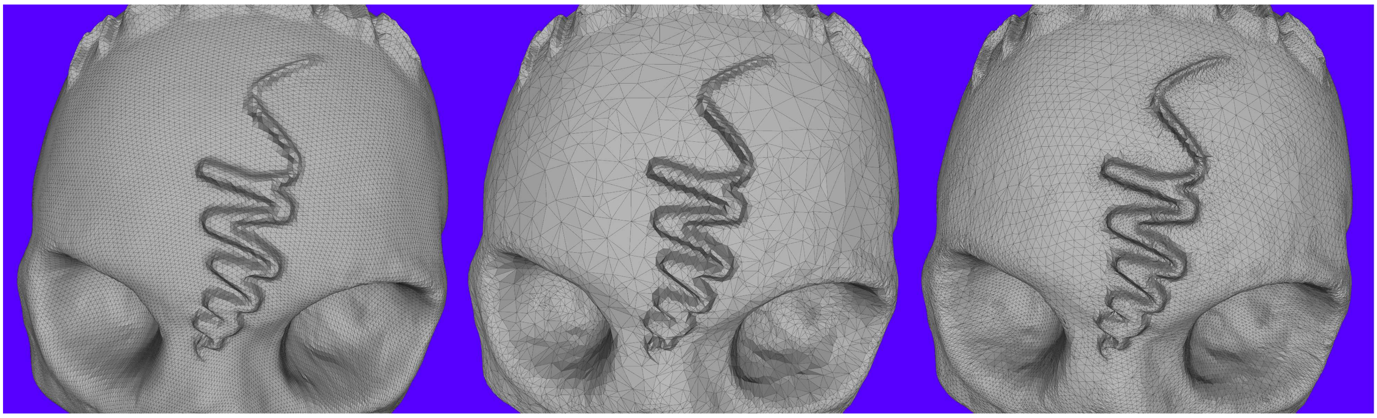


Fig. 10. Left: S uniformly remeshed with [10]. Middle: the original S . Right: S remeshed with our adaptive scheme. We observe that our scheme produces much less triangles in smooth regions while maintaining a better visual quality.

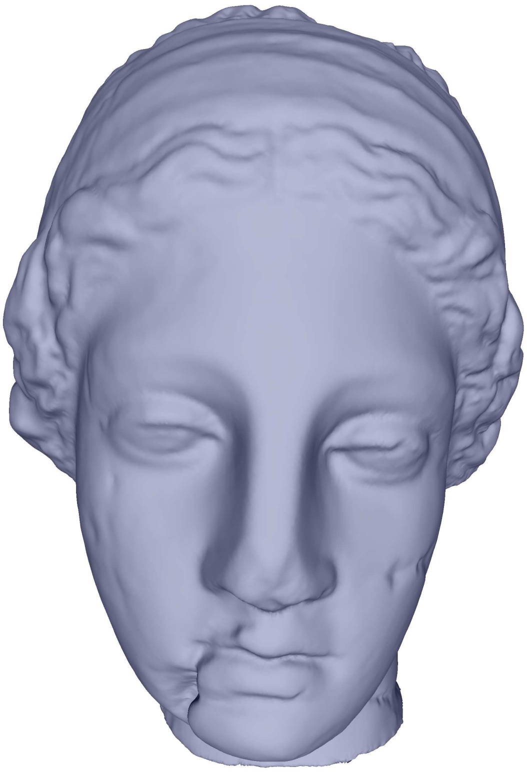


Fig. 11. V remeshed with our method.

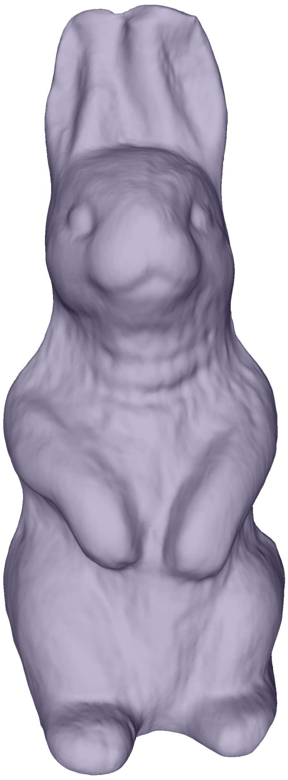


Fig. 12. R remeshed with our method.