Shape Operator Metric for Surface Normal Approximation

Citation	Cañas, Guillermo D. and Steven J. Gortler. 2009. Shape operator metric for surface normal approximation. In Proceedings of the 18th International Meshing Roundtable, October 25-28, 2009, Salt Lake City, Utah, ed. Brett W. Clark, Part 6, 447-461. Berlin: Springer.
Published Version	doi:10.1007/978-3-642-04319-2_26
Accessed	February 19, 2015 10:29:42 AM EST
Citable Link	http://nrs.harvard.edu/urn-3:HUL.InstRepos:9639956
Terms of Use	This article was downloaded from Harvard University's DASH repository, and is made available under the terms and conditions applicable to Open Access Policy Articles, as set forth at http://nrs.harvard.edu/urn-3:HUL.InstRepos:dash.current.terms-of-use#OAP

Shape Operator Metric for Surface Normal Approximation

Guillermo D. Canas and Steven J. Gortler

School of Engineering and Applied Sciences
Harvard University
33 Oxford St. Cambridge, MA.
{gdiez,sjg}@seas.harvard.edu

Summary. This work deals with the problem of practical mesh generation for surface normal approximation. Part of its contribution is in presenting previous work in a unified framework. A new algorithm for surface normal approximation is then introduced which improves upon existing ones in a number of aspects. In particular, it produces better approximations of surfaces both in practice and in the theoretical limit regime. Additionally, it resolves in a simple way some of the problems that previous methods for surface approximation suffered from.

1 Introduction

Computing high-quality approximating meshes from surfaces is an important problem in computational geometry, with many practical implications. Although the approximation criteria can vary greatly, often, approximating either surface position, or a surface's normal field can be a good criteria in practice. As has been argued elsewhere [7, 14], approximating a surface while minimizing normal approximation error can be useful in many applications.

There is a considerable body of previous work that deals with the surface approximation problem. Some notable examples include ϵ-nets [6], for surface and normal approximation, the Quadric Error Metric algorithm (QEM) [9] for surface approximation, and Variational Shape Approximation (VSA) [14], for surface and normal approximation.

In this paper, it is first discussed how the above three methods can be interpreted from within a unified framework. In this interpretation, they are essentially all minimizing a k-means like energy, where only the distance metrics are different. Interestingly enough, in the limit, for smooth surfaces, these distance measures converge to each other. The other key difference that is explored is how these means are used by the different algorithms to produce the output triangulation.

Next, a novel distance measure is proposed (Shape Operator Metric, or SOM), with a corresponding algorithm, that fills a natural gap in this framework. In particular, like VSA, it is designed for normal approximation. But, like QEM, it does not require a region-triangulation step. Such a step can complicate the implementation and, as it is discussed, it introduces a constant factor of inefficiency close to 2, in the limit of approximation.

2 Framework

Considered here are meshing algorithms for surface approximation that try to either meet a uniform error bound, or minimize the average approximation error over a surface M (minimizing error in the $\mathcal{L}^\infty$ or $\mathcal{L}^2$ sense respectively). These kinds of algorithms can be naturally described as an optimization problem:

$$\operatorname*{argmin}_{\{p_j\},\mathcal{V}_j} E^\infty_{\mathrm{x}} = \operatorname*{argmin}_{\{p_j\},\mathcal{V}_j} \max_j \max_{p\in\mathcal{V}_j} D_{\mathrm{x}}(p_j, p) \tag{1}$$

$$\operatorname*{argmin}_{\{p_j\},\mathcal{V}_j} E^2_{\mathrm{x}} = \operatorname*{argmin}_{\{p_j\},\mathcal{V}_j} \sum_j \int_{\mathcal{V}_j} D_{\mathrm{x}}(p_j, p)dp \tag{2}$$

over both a set of *means* $\{p_j\}$ (points on the surface), and a corresponding partition $\{\mathcal{V}_j\}$ of M composed of the Voronoi cells of $\{p_j\}$ with respect to a chosen distance function D_x.

Optimal Voronoi partitions have in all (except perhaps the rarest) cases neither the shape nor the topology of a triangle mesh. Some further step is generally necessary before producing a triangle mesh as output. In the sequel, a meshing algorithm is referred to as a *primal algorithm* if it discretizes the boundaries of Voronoi cells, triangulates their interior, and outputs this set of triangles, as in [14]; while an algorithm in which the means are instead directly connected using the dual topology of the partition $\{\mathcal{V}_j\}$ to produce a triangle mesh, as in [6], will be denoted as a *dual algorithm.*

Apart from the added algorithmic complexity, primal algorithms have an inherent approximation inefficiency in the limit. Roughly speaking, in smooth surface regions, in the limit, the relative sizes and aspect ratios of the Voronoi regions are optimized by minimizing the above energies. These relative sizing and aspect ratios will be maintained under mesh duality. But these sizes and aspect ratios are altered (within a constant factor) when the Voronoi regions are triangulated. The limit regime is explored in more detail in Appendix B, while the non-limit case is discussed experimentally in Section 5.

The algorithms of [6, 9, 14], as well as the one introduced in Sec. 3, all fit into this framework. In particular, the method in [14] introduced the idea of directly optimizing energies with the above form using a k-means/Lloyd-Max type algorithm. It then applies a primal meshing step to the resulting partition.

In the work of [6] one finds a set of means $\{p_j\}$ with bounds on the energy of Eq. 1. The means are then connected in a dual triangulation.

The QEM method of [9] applies a sequence of edge collapses to the input mesh, which can essentially be interpreted as an attempt to minimize 2. In particular, upon completion of the QEM algorithm, each vertex of the output triangulation can be thought of as a mean with a region of the surface associated to it: the portion of the surface that it uses to evaluate its associated (quadric) error (with adjacent regions slightly overlapping). In this sense QEM can be considered a dual algorithm. Even though the connectivity of the triangle mesh is not directly related to the Voronoi regions of Eq. 1, its limit behavior is analogous.

2.1 Surface Approximation

The algorithms of [6, 9, 14] use the following distances when optimizing Eqs. 1 or 2:

$$D_{\textsc{ii}}(p_j, p) = \min_{\gamma \in P(p_j, p)} \int_\gamma q^c_{\textsc{ii}}(\gamma'(t); \gamma(t))^{\frac{1}{2}} dt \tag{3}$$

$$D_{\textsc{qem}}(p_j, p) = < p - p_j, n(p) >^2 \tag{4}$$

$$D_{\textsc{svsa}}(p_j, p) = < p - p_j, n(p_j) >^2 \tag{5}$$

where $q^c_{\textsc{ii}}(\gamma'(t); \gamma(t))$ is the "convexified" (using the absolute value of the eigenvalues) second fundamental form at point $\gamma(t)$ and applied in direction $\gamma'(t) \in T_{\gamma(t)}M$, and $P(p_j, p)$ is the set of all paths that connect p_j to p on the surface.

As described in Appendix A, for smooth surfaces, it is possible to write: for all $\lambda > 0$, for all non-parabolic $p_j \in M$ there is an open neighborhood V of p_j such that $\forall p \in V$:

$$D_{\textsc{qem}}(p_j, p) \simeq_\lambda D_{\textsc{ii}}(p_j, p)^4 \simeq_\lambda D_{\textsc{svsa}}(p_j, p) \tag{6}$$

where the notation $\simeq_\lambda$, borrowed from [6], implies tight approximation to any desired degree of accuracy. Note that the exponent 4 above arises from the fact that ϵ-nets minimize a form of Euclidean distance between the surface and the approximation, while QEM and $sVSA$ minimize *squared* Euclidean distance. Equation 6 is valid only for elliptic points p_j. For hyperbolic points, $D_{\textsc{qem}}$ and $D_{\textsc{svsa}}$ still converge to the same value, but $(D_{\textsc{ii}})^4$ is only an upper bound of $D_{\textsc{qem}}$ and $D_{\textsc{svsa}}$, [Note that one could have defined $D_{\textsc{ii}}$ using $|q_{\textsc{ii}}|^{1/2}$ in the integrand of 3, where $q_{\textsc{ii}}$ is the second fundamental form. This would make $(D_{\textsc{ii}})^4$ be a tight approximation of $D_{\textsc{qem}}$ and $D_{\textsc{svsa}}$ everywhere non-parabolic on M, but would no longer be a Riemannian metric.]

The distance $D_{\textsc{ii}}$, is too expensive to compute in practice (since each evaluation involves computing a shortest path under the $q^c_{\textsc{ii}}$ surface metric). In contrast, both $D_{\textsc{qem}}$ and $D_{\textsc{svsa}}$ are efficiently computed using only local information at the arguments p_j and p.

2.2 Normal Approximation

The problem of computing a mesh that approximates the normal field of a surface is considered next. It is noted that for this problem, the normals of the approximating mesh are piecewise constant. However, instead of being inferred from the vertex positions, the normals of the output mesh are optimally *assigned* to triangles. This distinction is necessary to avoid difficulties like those described in [5, 7] that can occur when triangles have large internal angles, even if they have the right limit shape and size.

A similar analysis to that of Sec. 2.1 can be made in this case. Here, the two relevant algorithms that are considered are [6, 14]. They use the following distances to optimize Eqs. 1 and 2 respectively:

$$D_{\mathrm{III}}(p_j, p) = \min_{\gamma \in P(p_j, p)} \int_{\gamma} q_{\mathrm{III}}(\gamma'(t); \gamma(t))^{\frac{1}{2}}\, dt \tag{7}$$

$$D_{\mathrm{nVSA}}(p_j, p) = \|n(p_j) - n(p)\|^2 \tag{8}$$

where q_{III} is the surface's third fundamental form, and $n : M \to \mathbb{S}^2$ is the Gauss map.

Analogously as proven in Appendix A, it is, for p in an appropriate, small enough neighborhood of a non-parabolic p_j:

$$D_{\mathrm{III}}(p_j, p)^2 \simeq_{\lambda} D_{\mathrm{nVSA}}(p_j, p) \tag{9}$$

2.3 Behavior

To aid in our discussion, three different kinds of regions on a surface will be considered, and the algorithms under consideration evaluated separately for each. The following distinct types of regions on surfaces are considered:

In smooth and non-parabolic regions, it can be shown that, in the limit, the regions of the partitions generated by optimizing D_{sVSA} D_{nVSA} and D_{QEM} have the proper aspect ratio [10, 14], which is a necessary condition for optimality for their respective surface or normal approximation problem. It can also be shown that, for everywhere-elliptical surfaces, and in the limit, the method of ϵ-nets [6] using D_{II} produces results that are within a constant factor of the globally optimal $\mathcal{L}^\infty$ minimizer for the surface surface approximation problem. As discussed in Appendix B, in the limit, primal algorithms such as VSA will need roughly twice as many triangles as compared to dual algorithms such as QEM and ϵ-nets.

Near sharp features, these algorithms behave quite differently. In particular, one can see that D_{QEM} measures error “from the viewpoint” of the variable of integration $p \in M$, while D_{sVSA} does so from the viewpoint of the mean p_j. As a result, QEM places means at high-curvature points, and thus is suited as a dual algorithm, while sVSA (the surface approximation version of VSA [14]) places them at *low*-curvature points, making it better suited as a primal algorithm. nVSA also tends to place means at low-curvature points. It is less clear the authors how ϵ-nets behaves in this regard.

Parabolic/curved regions are places on a smooth surface near, or at parabolic points where there is significant higher-than-second-order bending (i.e. the surface is not locally well approximated by a quadratic patch), such as near the parabolic line on a torus. Algorithms often need special care in this case. Near parabolic/curved points, an optimization using D_{QEM} undersamples regions near curved parabolic lines. The original QEM algorithm [9] deals with this case by introducing special rules to prevent flips before edge collapses (which strictly-speaking breaks the energy-minimization formulation of Eq. 2). For ϵ-nets, an additional *isotropic* term is added to the distance to cope with such regions. VSA deals with this case, in which the Voronoi cell boundaries are highly curved, by discretizing these boundaries and triangulating the cells finely enough as to avoid undersampling.

3 Shape Operator Metric for Normal Approximation

An obvious missing piece in this description is an algorithm that converges to Eq. 9 in the limit, but places means at high curvature points away from it, making it most suitable as a dual algorithm. In some sense this algorithm would be to nVSA what QEM is to sVSA. Moreover it can be efficiently computed an optimized (as in Eq. 1 or 2), has high approximation efficiency in the sense of Appendix B, and it avoids heavy undersampling near curved parabolic lines.

To begin, consider the definition $D_{\text{nVSA}}(p_j, p) = \|n(p_j) - n(p)\|^2$, which measures normal error from either p_j or p, and, similarly as QEM, construct an approximation that only depends on p_j but not on any higher-order local information at p_j. To do this, a second-order Taylor expansion of $D_{\text{nVSA}}(p_j, p)$ around p is constructed (note that the zero-th and first order terms vanish):

$$D_{\text{SOM}}(p_j, p) \equiv (p_j - p)^T \frac{\partial D_{nVSA}(p'_j, p)}{\partial^2 p'_j}(p_j - p) \tag{10}$$

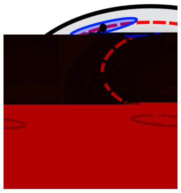
$$= (p_j - p)^T S(p)^2 (p_j - p) \tag{11}$$

where $S(p)$ is a $\mathbb{R}^{3\times 3}$ shape operator matrix $S(p) = k_1(p)e_1(p)e_1(p)^T + k_2(p)e_2(p)e_2(p)^T$, $\{k_1, k_2\}$ are the principal curvatures, and $\{e_1, e_2\}$ the principal directions.

Note that D_{SOM}, like D_{QEM} and D_{VSA}, can be efficiently computed only from local information at the endpoints, and, as will be shown in Sec. 4, results in an energy of the type of Eq. 1 or 2 that can be efficiently minimized using standard algorithms [13, 12].

The SOM algorithm then simply outputs the dual trianglulation of this computed surface partition.

It follows from the fact that this is a dual algorithm, whose distance converges in the limit to that of D_{nVSA}, and from the discussion of Appendix B, that this algorithm has the desired favorable (limit) efficient approximation characteristics when compared to the primal algorithm of [14]. It is also shown in Appendix C that this algorithm produces elements that conform to the theoretically optimal limit shape and orientation.



Unlike [6, 9, 14], curved parabolic regions are dealt with in a natural way, without special consideration, which adds to the simplicity of the algorithm. The side figure illustrates this point, where a mean p_j is placed at a parabolic line (red). Because the parabolic line is curved, p_j does not lie along the flat direction when viewed from the point of view of nearby points p. An SOM primal-region centered around p_j thus cannot grow too much along the parabolic line if the parabolic line curves.

It is possible to see that minimizing Eq. 2 using Eq. 10 has the effect of placing means at points of high-curvature. Consider the closely-related problem of gradient approximation of a scalar function f defined on the plane, and an analogous distance $D_{\text{fSOM}} = (p_j - p)^T H_f^2(p)(p_j - p)$ with $p_j, p \in \mathbb{R}^2$, where H_f is the Hessian of f. In an everywhere-isotropic region, $D_{\text{fSOM}} = k(p)\|p_j - p\|^2$, which, used for $\mathcal{L}^2$ minimization in a form analogous to Eq. 2 over the plane, is an instance of the *weighted k-means* problem, which is well-known to place means at points with high weight [1] (high-curvature in this case). The case where H is not isotropic behaves similarly, but the weight can be thought of as directionally-varying.

4 Implementation of SOM

The energy of Eq. 10 is minimized in a way very similar to the algorithm of [12], which uses a probabilistic seeding of means, followed by a Lloyd relaxation [13] and has theoretical guarantees of closeness to the global optimum. In this work, the probabilistic seeding is simply replaced by iteratively placing means at the surface point with maximum minimum-distance to the current set of means, similarly as the greedy algorithm for computing ϵ-nets of [11]. This is also similar to the optimization method of [6], except that the seeding is followed by a Lloyd relaxation, and is also similar to [14]. The shape operator matrix S of Eq. 10 is estimated using the algorithm of [3].

Once the seed means have been placed, the Lloyd relaxation has two stages. The first creates a distance-dependent Voronoi partition of the surface, and the second computes the new means' locations from the current partition.

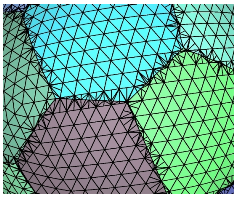
To compute a Voronoi partition, all vertices (as opposed to input triangles) are tagged as belonging to some primal Voronoi region, and Voronoi region boundaries are computed by splitting input triangles that have vertices in different regions, as in the side figure below. A Voronoi region is thus not constrained to be a collection of faces, but can have a boundary that cuts across triangles, which may slightly improve accuracy in practice. Also, in this way, Voronoi regions can meet at most at 3-way junctions. These 3-way junctions naturally dualize into triangles. Note that this generalizes to higher-dimensions, so that, by construction, it will only output simplicies.

Given a Voronoi partition of the surface, the new means' locations are computed. First, note that the energy of Eq. 2 for the distance D_{SOM} can be written as

$$E^2_{\text{SOM}} = \sum_j p_j^T \Big(\int_{\mathcal{V}_j} S(p)^2 dp \Big) p_j - \tag{12}$$

$$- 2p_j^T \Big(\int_{\mathcal{V}_j} S(p)^2 \cdot p dp \Big) + \Big(\int_{\mathcal{V}_j} p^T \cdot S(p)^2 \cdot p dp \Big) \tag{13}$$

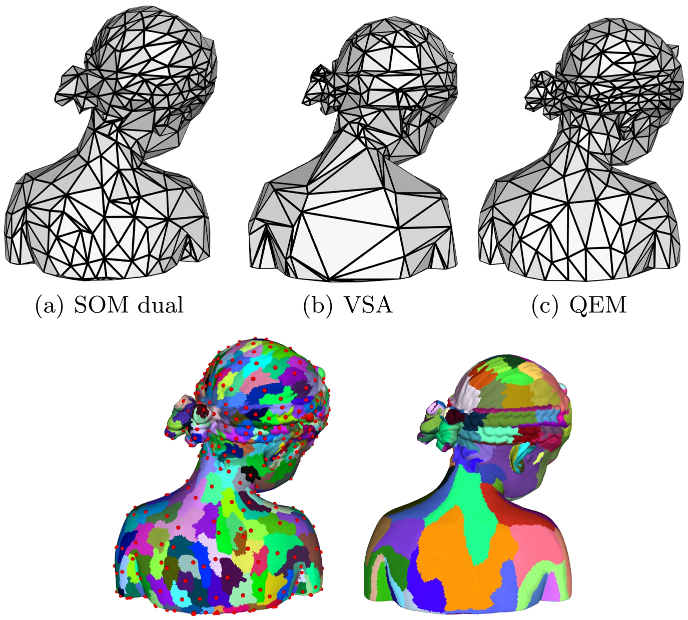
and so it is quadratic in p_j.



It is possible to compute the minimizer p_j of Eq. 12 by solving a small linear system, but this would return a mean p_j which is not constrained to be on the surface. Instead, the constants in equation 12 are computed in a first pass: $A_j = \int_{\mathcal{V}_j} S(p)^2 dp$ and $b_j = \int_{\mathcal{V}_j} S(p)^2 \cdot p dp$, for each Voronoi region $\mathcal{V}_j$. Then, for each input triangle (or split triangle) inside region $\mathcal{V}_j$ the barycentric coordinates (u, v) of the minimizer p_j of Eq. 12 can be found by solving $R^T A_j R \begin{pmatrix} u \\ v \end{pmatrix} = R^T b_j$ where $R \in \mathbb{R}^{3\times 2}$ is some basis of the supporting plane of the triangle. The minimizer may fall outside the triangle, so it is necessary to look for it along triangle edges and vertices as well. The final mean is the minimum over all the minimizers on each triangle, guaranteeing that p_j is a point on the surface. Finally, instead of outputting p_j directly as a (dual) vertex, a quadric error metric [9] for its associated Voronoi region $\mathcal{V}_j$ is first computed, and its minimizer along the line passing through p_j in direction $n(p_j)$ is output. This small perturbation slightly improves the approximation.

5 Results

Some surfaces processed by the SOM algorithm are shown in figures 1 through 3. These meshes are compared with those produced by VSA, which are computed by exactly following [14]. Note that, unlike SOM, VSA has a free parameter (the precision used to discretize the partition regions's boundaries), which has been tuned to improve VSA's output. These results are also compared with the output of QEM [9]. Note that QEM optimizes (RMS) distance from the surface to the approximation, instead of normal error, and therefore



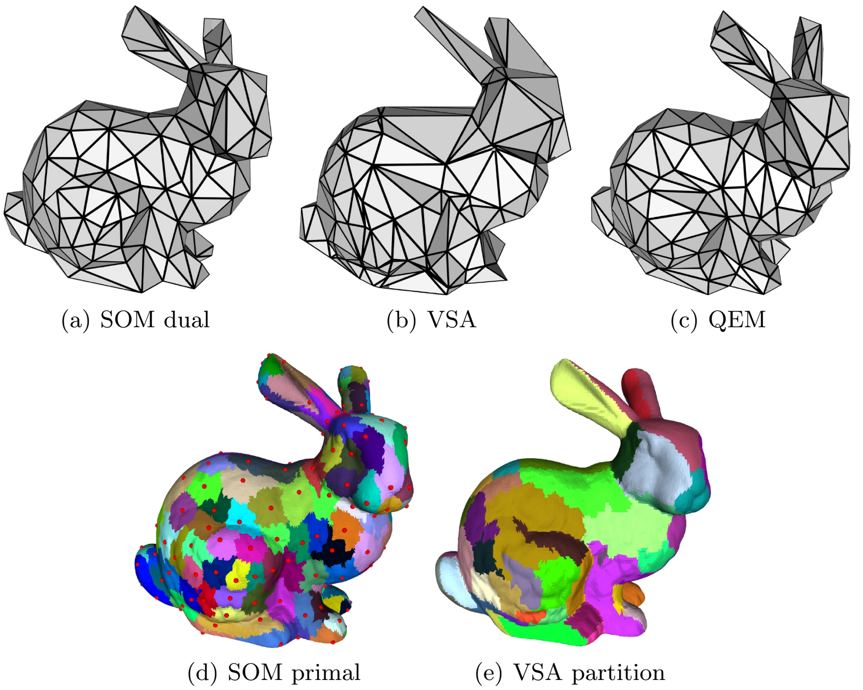
(a) SOM dual (b) VSA (c) QEM

(d) SOM primal (e) VSA partition

Fig. 1. SOM: 500 vert., 996 tris. (Hausdorff error = 1.79e-2, RMS error = 2.24e-3). **VSA**: 528 verts., 1076 tris. (Hausdorff error = 2.31e-2, RMS error = 5.06e-3). — **QEM**: 502 verts., 1000 tris. (Hausdorff error = 1.93e-2, RMS error = 1.93e-3).

the comparison is not strictly relevant; but it is included it as reference. Runtimes for SOM range from 5 sec. (bunny, input: 70k tris.) to 40 sec. (statue, input: 512k tris.), on a single core 2.0GHz CPU.

Even though it is not necessarily what is being optimized for in this work, it is possible to consider ($\mathcal{L}^\infty$) Hausdorff, and RMS error in the sense of surface approximation. Note that, in most cases, QEM produces slightly better approximation of the surface than SOM, and significantly better than VSA. This is expected, as QEM optimizes surface approximation error (RMS error in the figures), while VSA and SOM both optimize normal error instead. Notice that, for smooth surfaces, and using (almost) the same number of triangles, SOM's approximation is appreciably finer than VSA's. On smooth surfaces, the approximation is significantly better for SOM at a given sampling rate. As can be seen in the primal partitions in figures 1 and 2, with an equal triangle budget, SOM is able to partition the surface into smaller regions that capture detail better. Note that the bunny is particularly troublesome for VSA, when compared with SOM, because its bumpy surface produces very curved regions that can output many triangles when their boundaries are discretized by the VSA algorithm. In general, in the above figures, triangles are elongated along the directions of minimum curvature, and tend to show



(a) SOM dual

(b) VSA

(c) QEM

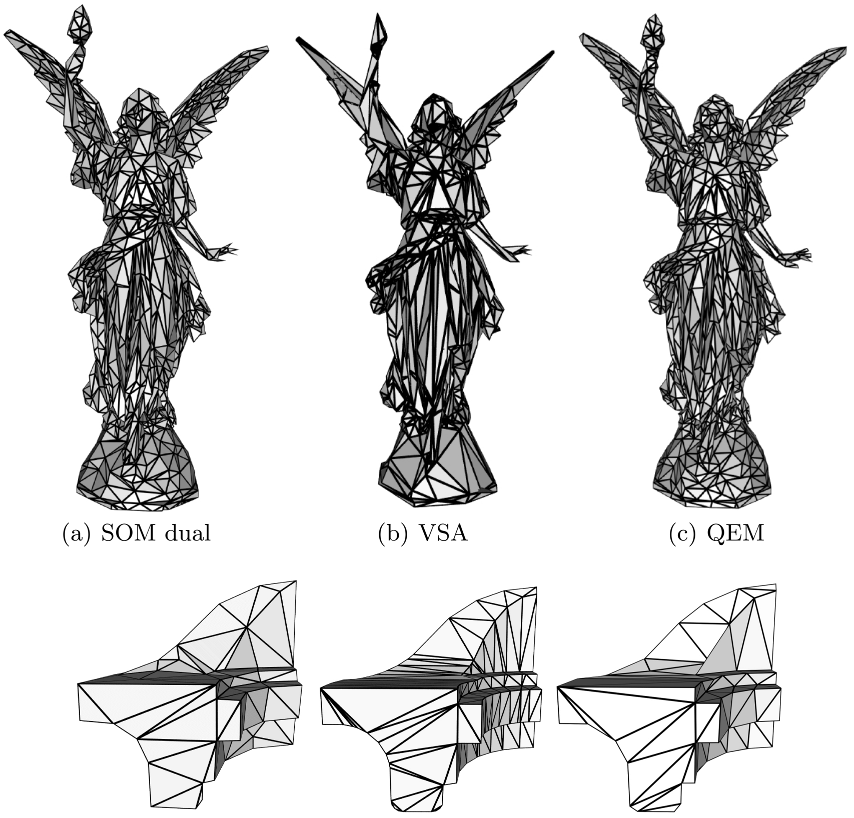
(d) SOM primal

(e) VSA partition

Fig. 2. **SOM**: 200 verts., 396 tris. (Hausdorff error = 3.32e-1, RMS = 5.02e-2). **VSA**: 199 verts., 409 tris. (Hausdorff error = 7.03e-1, RMS error = 9.93e-2). — **QEM**: 202 verts., 400 tris. (Hausdorff error = 2.50e-1, RMS error = 4.31e-2).

very high anisotropy in places where this is possible: like the ears of the bunny or the statue's arms. Note that our algorithm offers no guarantees in terms of normal flips in the triangulation, which could show up occasionally in sparsely sampled regions. This behavior is similar to VSA and ϵ-nets, which also cannot guarantee to be free of flips.

Figure 3(d-f) shows a surface composed of roughly flat parts separated by sharp features. On these kinds of surfaces VSA does particularly well, since it operates by locating roughly-flat patches and triangulating them. In particular, the region-triangulation phase of VSA is well-tuned to this problem, since the desired behavior in this case is to triangulate the flat polygons. SOM in this case naturally places means at sharp corners. But its connectivity is guided by the shape operator, which is almost everywhere degenerate here. This case is dealt with by computing the final mesh connectivity using a modified shape operator, which is set to a very high (isotropic) value in flat regions, effectively simulating a flat-polygon triangulation step (similar to the constrained Delaunay triangulation used in [14]).



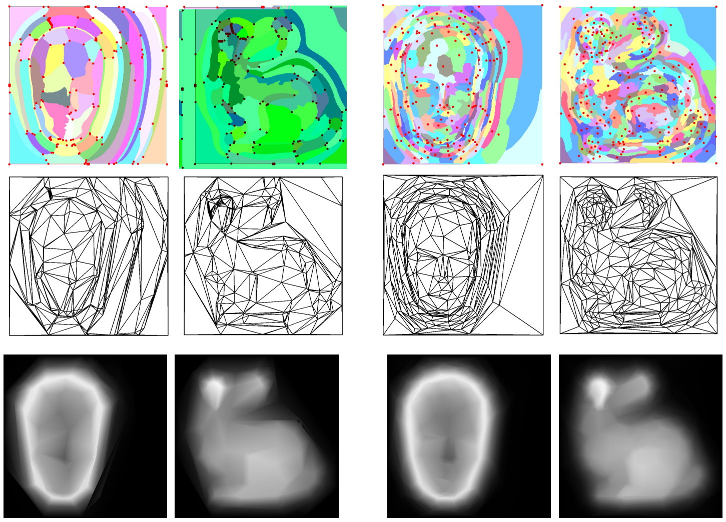
(a) SOM dual (b) VSA (c) QEM

(d) SOM dual (e) VSA (f) QEM

Fig. 3. Lucy SOM: 1500 verts., 2988 tris. (Hausdorff = 14.598, RMS = 1.866). **Lucy VSA**: 1456 verts., 2990 tris. (Hausdorff error = 44.688, RMS error = 5.911). **Lucy QEM**: 1496 verts., 2988 tris. (Hausdorff error = 11.834, RMS error = 1.472). **Fandisk SOM**: 80 verts., 156 tris. (Hausdorff error = 0.118, RMS error = 0.0157). **Fandisk VSA**: 80 verts., 156 tris. (Hausdorff error = 0.0596, RMS error = 0.0131). **Fandisk QEM**: 80 verts., 156 tris. (Hausdorff error = 0.264, RMS error = 0.0152).

5.1 Numerical validation

Unlike for surface approximation, there is, as far as the authors are aware, no standard way of measuring the surface *normal* approximation on a surface. If there was, away from the limit regime, a well-defined one-to-one correspondence between points on the surface and points on the approximation, then it would be possible to compute the ($\mathcal{L}^2$) approximation error by integrating the distance between corresponding normals over the surface. However, this correspondence is not available. To analyze approximation error, the very closely-related problem of approximation of the gradient of a height field over



(a) nVSA (b) SOM

Fig. 4. Gradient approximation using nVSA and our algorithm. Primal (top), algorithm output (middle), height field approximation (bottom). Red marks in the primal are locations of vertices in the dual. In both (a) and (b), the mask (left side) is approximated with 356 triangles, and the bunny (right side) with 468 triangles.

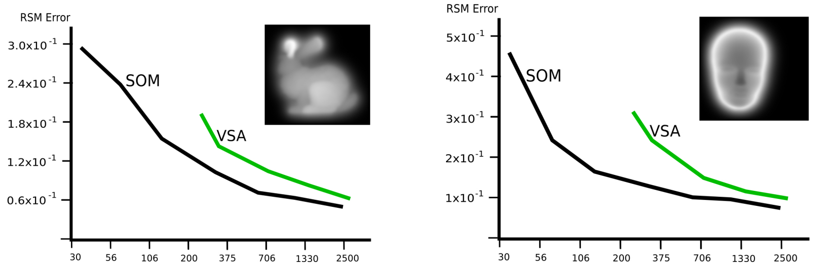


Fig. 5. RMS ($\mathcal{L}^2$) gradient error plots for the mask and bunny height fields (top-right corners.)

the plane is considered [it has optimal limit aspect ratio ξ_1/ξ_2, where ξ_i are the eigenvalues of the heigh field Hessian [4, 5]]. Because both VSA and SOM only look at normals and shape operators, it is possible to naturally adapt both to the gradient approximation case by measuring distances between gradients, as opposed to normals, by computing a Hessian of the height field at each point, instead of a shape operator. Both algorithms must also be extended to force them to conform to the boundary of the domain. There is however, to our knowledge, no equivalent natural generalization of QEM [9] to the heigh field approximation case. Once again, the tunable parameter in VSA has been adjusted to the best results obtained. The input is a surface that is finely scan-converted on a squared piece of the plane (figure 5 top-right corners.) Planar meshes obtained this way are shown in figure 4, while figure 5 shows the corresponding error plots for these two inputs, at several approximation levels. Notice how the mesh approximating the mask in 4.b more closely matches the features of the input than 4.a, even though both have the same number of triangles. The difference in RMS error is not always large for a fixed number of triangles, though it is significant. If, alternatively, an RMS error level is fixed, and the VSA and SOM approximations with that error are considered, it can be noted that the SOM mesh has significantly fewer triangles.

6 Summary and Conclusion

This work begins by placing some established algorithms for surface approximation into a common framework. From this analysis, it becomes apparent that a *dual* variational algorithm for surface normal approximation was previously missing. Such algorithm is introduced next, and its limit behavior shown to conform with the theoretical asymptotic aspect ratio (Appendix C). It is further argued that this dual algorithm has several advantages over primal variational algorithms for surface normal approximation (such as VSA). In particular, the limit approximation efficiency is discussed in Appendix B, which is shown to be approximately 1.75 times higher for a dual algorithm with the same (asymptotically optimal) limit aspect ratio. The approximation results of the proposed algorithm and established ones are also compared on practical data sets. While the primal VSA is still preferable for piecewise flat surfaces, where it successfully splits them into flat regions which are then triangulated, for general curved surfaces, the algorithm introduced in this paper is shown to perform better. This is further shown on quantitatively experiments, which are carried out on the very closely related problem of gradient approximation over the plane.

References

1. Fujiki Morii (2006). A Generalized K-Means Algorithm with Semi-Supervised Weight Coefficients. ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition. 198–201.
2. P. M. Gruber (2004). Optimum quantization and its applications. Adv. Math. 186. 456497.
3. Szymon Rusinkiewicz (2004). Estimating Curvatures and Their Derivatives on Triangle Meshes. Symposium on 3D Data Processing, Visualization, and Transmission.
4. E. F. D'Azevedo and R. B. Simpson (1991). On Optimal Triangular Meshes for Minimizing the Gradient Error. Numerische Mathematik, 59. 321–348.
5. Guillermo D. Canas and Steven J. Gortler (2006). On Asymptotically Optimal Meshes by Coordinate Transformation. 15th International Meshing Roundtable.
6. Kenneth L. Clarkson (2006). Building Triangulations Using Epsilon-Nets. STOC: Proceedings of the Thirty-eighth Annual SIGACT Symposium.
7. Jonathan R. Shewchuk (2002). What Is a Good Linear Finite Element? - Interpolation, Conditioning, Anisotropy, and Quality Measures. Eleventh International Meshing Roundtable. Ithaca, New York. 115–126.
8. E. Nadler (1986). Piecewise linear best l2 approximation on triangulations. Approximation Theory V. 499–502.
9. Michael Garland and Paul Heckbert (1997). Surface Simplification Using Quadric Error Metrics. ACM SIGGRAPH.
10. Heckbert,, Paul S. and Garland,, Michael (1999). Optimal triangulation and quadric-based surface simplification. Comput. Geom. Theory Appl., 14, 1-3. 49–65.
11. T. Gonzalez (1985). Clustering to Minimize the Maximum Inter-Cluster Distance. Theoretical Computer Science, 38. 293–306.
12. David Arthur and Sergei Vassilvitskii (2007). K-means++: the advantages of careful seeding. SODA: Symposium On Discreet Algorithms. 1027–1035.
13. Stuart P. Lloyd (1982). Least squares quantization in PCM. IEEE Transactions on Information Theory, 28, 2. 129–136.
14. David Cohen-Steiner and Pierre Alliez and Mathieu Desbrun (2004). Variational shape approximation. ACM SIGGRAPH.

Appendix A: Distance tightness bounds

As in [6], it is said that $a \leq_\lambda b \Leftrightarrow a \leq (1+\lambda)b$, and that $a \simeq_\lambda b \Leftrightarrow a \leq_\lambda b \wedge b \leq_\lambda a$. $q_{\mathrm{II}}(t,p)$ is the second fundamental form at point p in direction t, and q^c_{II} is its "convexified" version (taking the absolute value of the eigenvalues). The definitions of D_{II} and D_{III} are in Eqs. 3 and 7.

Lemma 1 *For all $\lambda > 0$, for all non-parabolic $p_j \in M$, there's an open neighborhood $V \ni p_j$ of M such that $\forall p \in V$, $D_{QEM}(p_j,p) \leq_\lambda D_{II}(p_j,p)^4$, and $D_{sVSA}(p_j,p) \leq_\lambda D_{II}(p_j,p)^4$.*

Proof. Lemma 4.1 of [6] shows that for all $\lambda > 0$, for all non-parabolic $p_j \in M$, there's an open neighborhood $V \ni p_j$ of M such that $\forall p \in V$, $D_{\mathrm{E}}(p_j, T_pM) \leq_\lambda D_{\mathrm{II}}(p_j,p)^2$, where $D_{\mathrm{E}}(p_j, T_pM)$ is the Euclidean shortest distance from p_j to the plane tangent to the surface at p. Then, by the symmetry of D_{II}:

$$\begin{aligned} D_{\mathrm{QEM}}(p,p') = D_{\mathrm{E}}(p_j,p)^2 \leq_\lambda D_{\mathrm{II}}(p_j,p)^4 = \\ = D_{\mathrm{II}}(p,p_j)^4 \geq_\lambda D_{\mathrm{E}}(p,p_j)^2 = D_{\mathrm{sVSA}}(p_j,p) \end{aligned}$$

where V is chosen small enough such that the neighborhood $V' \ni p$ of the last approximate inequality above contains p_j as well.

Note that the other direction of the inequality is not true in general in neighborhoods that are not elliptic. If D_{II} had been defined using $|q_{\mathrm{II}}|^{1/2}$ instead, then it would've been possible to write: $D_{\mathrm{QEM}}(p,p') \simeq_\lambda D_{\mathrm{II}}(p_j,p)^4 \simeq_\lambda D_{\mathrm{sVSA}}(p,p')$ at every non-parabolic point.

Lemma 2 *For all $\lambda > 0$, for all non-parabolic $p_j \in M$, there's an open neighborhood $V \ni p_j$ of M such that $\forall p \in V$, $D_{SOM}(p_j,p) \simeq_\lambda D_{III}(p_j,p)^2$, and $D_{nVSA}(p_j,p) \simeq_\lambda D_{III}(p_j,p)^2$.*

Proof. From the fact that $D_{\mathrm{SOM}}(p_j,p)$ is a second-order Taylor approximation of $D_{\mathrm{nVSA}}(p_j,p)$ around p, and that p_j is not parabolic, with V chosen small enough not to contain parabolic points (which is possible since the set of non-parabolic point is open) follows that $\forall \beta > 0$ there's a neighborhood $V \ni p_j$ such that $\forall p \in V$, $D_{\mathrm{nVSA}}(p_j,p) \simeq_\beta D_{\mathrm{SOM}}(p_j,p)^2$ for $0 < \beta < \lambda, (1+\beta)^2 = 1+\lambda$. It is also possible to choose a neighborhood $V' \ni p_j$ small enough such that [2]:

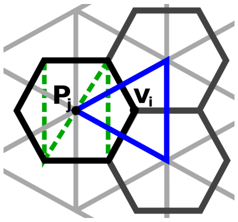
$$D_{\mathrm{SOM}}(p_j,p) = (p_j - p)^T S^2(p)(p_j - p) \simeq_\beta D_{\mathrm{III}}(p_j,p)^2$$

In particular, because $\beta < \lambda$, then also $D_{\mathrm{SOM}}(p_j,p) \simeq_\lambda D_{\mathrm{III}}(p_j,p)^2$. Finally, inside the intersection of the two neighborhoods from the two claims, the transitivity property $x \leq_\beta y \leq_\beta z \Rightarrow x \leq (1+\beta)^2 z$ yields $D_{\mathrm{nVSA}}(p_j,p) \simeq_\lambda D_{\mathrm{III}}(p_j,p)^2$.

Appendix B: Limit approximation efficiency

As pointed out in [6], an optimal solution of Eq. 1 (or 2), in the limit regime, for a small enough, regular (everywhere elliptical or hyperbolic) neighborhood of a surface point, looks like a (stretched) regular hexagonal tiling. A dual algorithm outputs the dual of this tiling (blue), which locally is a regular (valence 6) triangulation. A primal algorithm instead triangulates the hexagons directly (green). The limit efficiencies of these dual and primal triangulations are compared next.

The uniform stretching is first undone to obtain an isotropic hexahedral decomposition, which can be shown not to affect the analysis. Note that, although there are multiple ways of triangulating a regular hexagon, all produce the same set of triangles if symmetry and rotation are discounted. In the $\mathcal{L}^\infty$, normal approximation case, the larger triangles of the primal (green) have error equal to $D_{\mathrm{III}}(p_j, v_i)$, same as the error of the dual triangles (and analogously for surface approximation by using D_{II} instead). But there are four primal triangles per hexagon, and only two dual triangles per hexagon, resulting in a factor of two inefficiency of the primal.



The $\mathcal{L}^2$ case is more involved, and it is only analyzed for the normal approximation case that concerns us most here. Optimal normals are *assigned* to each triangle in both triangulations, which can be computed in closed-form. The $\mathcal{L}^2$ normal error over the triangles is then numerically integrated. Starting from the same regular hexagonal tiling, here the error per unit area in the primal and the dual triangulations is different. Using the fact that $\mathcal{L}^2$ error grows as s^4 where s is a uniform scale factor applied to the triangulation, it is possible to scale the dual triangulation until its error per unit area matches that of the primal. Now the average triangle areas can be compared, yielding the inefficiency factor between primal and dual. All computations (including integration and scaling) use conservative bounds. The limit inefficiency factor is $\gamma \in (1.7635, 1.7642)$ (where lower and upper bounds are rounded down and up, respectively). Hence a primal triangulation in the limit uses approx. 75% more triangles to obtain the same $\mathcal{L}^2$ normal error as the dual.

Appendix C: Shape operator metric and aspect ratio

For a regular (non parabolic) point p on a smooth surface M, and for very fine approximations, it is possible to consider the shape of a neighborhood $\mathcal{N}_p$ of fixed area that locally minimizes Eq. 2 using D_{SOM}. Since the neighborhood is very small and the surface is smooth, the shape operator is approximately constant inside. Therefore, to any desired degree of accuracy, $\mathcal{N}_p$ is the minimizer of $\int_{\mathcal{N}_p} (p'-p)^T \cdot S(p)^2 \cdot (p'-p) dp'$. If this expression is written in a frame centered at p and oriented so that $\hat{z} = n(p)$ and $\{\hat{x}, \hat{y}\}$ are the principal directions of $S(p)$, then this energy is $\int_{\mathcal{N}_p} k_1^2 x^2 + k_2^2 y^2 dxdy$, where k_1, k_2 are the principal curvatures at p. It is easy to show that a neighborhood $\mathcal{N}_p$ of fixed area minimizing this energy is an ellipse of aspect ratio (ratio of major to minor axis) k_1/k_2, which matches the asymptotically optimal aspect ratio for normal approximation of [4]. Note that (around elliptic points) this ratio would have been $|k_1/k_2|^{1/2}$ for the surface approximation energy $\int_{\mathcal{N}_p} |k_1|x^2 + |k_2|y^2 dxdy$, in accordance with [8]. The dual triangulation inherits these properties: in the limit regime, dual triangles have circumscribing ellipses with same orientation and aspect ratio as the primal regions.