Willmore energy vs area as criteria for mesh optimisation

Willie Brink*, Lyuba Alboul*

*Sheffield Hallam University, {w.brink,l.alboul}@shu.ac.uk

Abstract

We present a general framework for a mesh optimisation algorithm designed to minimise a given cost function by performing the edge flip operation successively. Three cost functions are defined based on surface area, Willmore energy and a new energy that attempts to localise Willmore energy around the edges. From experimental results we conjecture that an edge flip decreases the surface area if and only if the localised Willmore energy is decreased.

1 Introduction

Surface reconstruction is a fundamental problem in computer graphics that has received much attention in recent years (examples include [2, 6]). A solution typically involves fitting a triangulated mesh to an unorganised point set, followed by an optimisation procedure in which the initial triangulation is changed to fit the data in some specified optimal manner (or such that the polyhedral surface approximates the original smooth surface optimally).

This paper deals with optimising a given input mesh. We provide the general framework for an optimisation algorithm. The algorithm incorporates the edge flip operation due to its simplicity and the fact that it changes a mesh on a local scale which implies low computational cost.

The basic idea behind the algorithm has been used previously to minimise various energies defined on the geometry of a mesh such as surface area or curvature [1, 5].

We present three schemes that can be incorporated in the general algorithm. The schemes are based on surface area, Willmore energy and localised Willmore energy that adds temporary vertices to the mesh. Experimental results indicate a strong relationship between this localised Willmore energy and the minimisation of surface area.

2 Our optimisation algorithm

Consider a triangular mesh M on a fixed set of points in $\mathbb{R}^3$. We denote by V, E and F respectively the set of vertices, edges and faces of M.

Our optimisation algoritm incorporates the edge flip operation illustrated in Figure 1. For some edge $e = (v_i, v_j)$ with adjacent faces $v_i v_k v_\ell$ and $v_i v_j v_\ell$, this operation replaces e with the edge (v_k, v_ℓ) and consequently those two faces with $v_i v_k v_\ell$ and $v_j v_\ell v_k$.

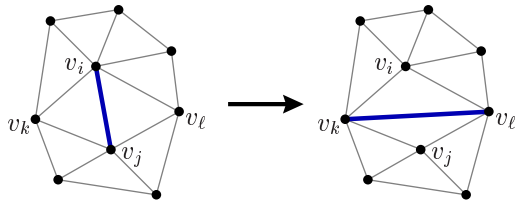

Figure 1: The edge flip operation on an edge (v_i, v_j).

The basic idea behind the optimisation algorithm is to flip edges successively in order to minimise some predefined cost function that typically measures some geometric property of a given mesh. For this purpose we calculate the reduction in the cost function for every edge e, i.e. the difference in the cost function before and after e is flipped. These reduction values, which we denote by $f(e)$ for each edge e, are used to determine the order in which edges are flipped.

A schematic overview of the general framework of our optimisation algorithm is given in Figure 2. We will denote by $M^{(0)}$ the initial input mesh and by $M^{(r)}$ the mesh resulting from flipping the first r edges in the algorithm.

In the first step we initialise r to 0 and calculate $f(e)$ for every edge e in $M^{(0)}$. These values will depend on the chosen cost function.

In step 2 an edge e^* is chosen such that $f(e^*)$ is maximum over all edges in $M^{(r)}$. Flipping this edge would result in a maximum reduction of the cost function.

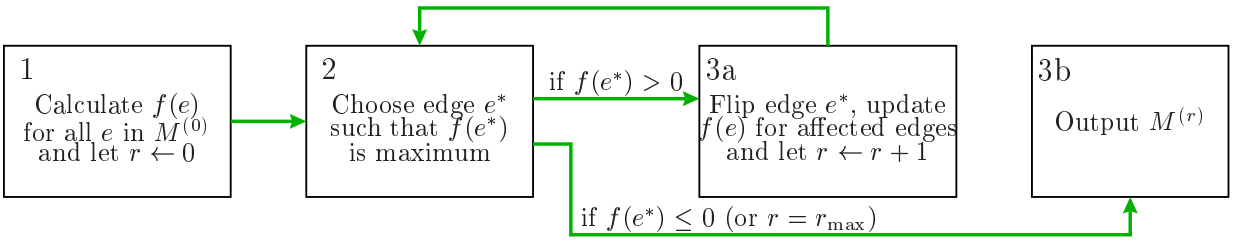

Figure 2: An overview of our general mesh optimisation algorithm on an initial mesh $M^{(0)}$.

If $f(e^*) \leq 0$ then flipping e^* would not decrease the cost function. Since $f(e^*)$ is maximum there are no edges that would result in a decreasing, and the algorithm stops.

If $f(e^*) < 0$, however, then the cost function decreases when e^* is flipped. We flip e^* and increase r by 1. We also update the reduction values of all edges affected by this edge flip. Since a single flip changes the mesh on a local scale it is usually necessary to update these values for only a few edges.

After successfully flipping an edge the algorithm jumps back to step 2 and repeats the process until no more edge flips can decrease the cost function (or, to avoid a possible infinite loop, until r reaches some user-specified value $r_{\max}$). The algorithm then outputs the current mesh $M^{(r)}$.

Note that it is quite possible that this could yield a mesh where the cost function is merely at a local minimum. Our experiments however do suggest that these local minima are fairly "good". Escaping local minima is currently under investigation.

During the optimisation process we require that the topological type of the mesh remain unchanged. To maintain this requirement we force the algorithm to disregard some edges. In particular an edge may not be flipped if its flipped counterpart is already an edge in the mesh (see for example Figure 1 — if the edge (v_k, v_ℓ) exist then the edge (v_i, v_j) should not be flipped). The edge flip operation is not defined for boundary edges and these should also remain unchanged. For the algorithm to ignore these particular edges we set the appropriate reduction values to $-\infty$, so that these edges would never be chosen as candidates for flipping.

The next section deals with defining cost functions that the algorithm can attempt to minimise.

3 Cost functions

The general framework of the algorithm described in the previous section has been used to minimise various different types of cost functions, such as area [8] and discrete analogues of known curvatures such as mean and total absolute curvature [1, 5].

In the following subsections a few different cost functions are defined. We will focus mainly on minimising Willmore energy [9] described in sections 3.2 and 3.3, and then compare results with the minimisation of surface area described next.

3.1 Area

This section describes how a reduction value function based on surface area may be defined.

Clearly, the total surface area A of a triangular mesh is given by the sum of areas of all triangular faces.

In order to minimise A with the algorithm described in section 2 it is necessary to define reduction values, which is denoted here by $f_a(e)$ for each edge e, that measures the difference in total area before and after a specific edge is flipped. Note from Figure 1 that an edge flip changes only the two faces $v_i v_k v_j$ and $v_i v_j v_\ell$ into $v_i v_k v_\ell$ and $v_j v_\ell v_k$. Hence

$$f_a(e) = a_{ikj} + a_{ij\ell} - a_{ik\ell} - a_{j\ell k}, \tag{1}$$

where a_{ijk} gives the area of the triangle with vertices v_i, v_j and v_k. If $f_a(e) < 0$ for an edge e then flipping e would reduce the total surface area.

In step 3a of our optimisation algorithm (see Figure 2) we update the reduction values of all edges affected by the edge flip, that is, all edges e for which $f_a(e)$ changes as a result of flipping e^*. Assuming $e^* = (v_i, v_j)$ and using the vertex labelling depicted in Figure 3, the reduction values of the following five edges should be updated as:

$$\begin{aligned}
f_a(e_{ij}) &\leftarrow -f_a(e_{ij}),\\
f_a(e_{ik}) &\leftarrow a_{itk} + a_{ik\ell} - a_{it\ell} - a_{k\ell t},\\
f_a(e_{i\ell}) &\leftarrow a_{ik\ell} + a_{i\ell u} - a_{iku} - a_{\ell uk},\\
f_a(e_{jk}) &\leftarrow a_{j\ell k} + a_{jkv} - a_{j\ell v} - a_{kv\ell},\\
f_a(e_{j\ell}) &\leftarrow a_{jw\ell} + a_{j\ell k} - a_{jwk} - a_{\ell kw},
\end{aligned}$$

where e_{ij} denotes the edge (v_i, v_j). The reduction values of the rest of the edges in the mesh remain unchanged.

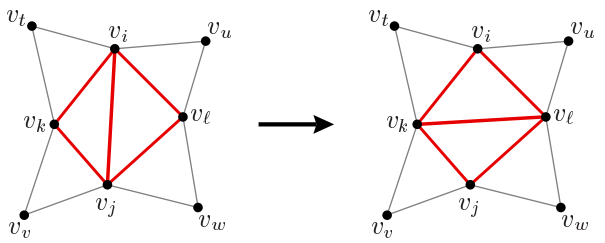

Figure 3: The neighbourhood of an edge (v_i, v_j) showing for which edges the area reduction values are affected by a flip.

Section 4 gives examples of applying the optimisation algorithm with this area criterion on a few models. Results are shown that suggest that minimising area may be closely related to minimising Willmore energy, the cost function described next.

3.2 Willmore energy

The next cost function to be minimised by our algorithm is the Willmore energy of a surface, which is a conformally invariant function of mean and Gaussian curvature.

Recently Bobenko [3], and in a later paper Bobenko and Schröder [4], introduced a discrete analogue of this energy and argued why surfaces with minimal Willmore energy are of importance. The discrete analogue is defined at each vertex v as

$$W(v) = \sum \beta(e) - 2\pi, \tag{2}$$

where the sum is taken over all incident edges of v. For each edge e the angle $\beta(e)$ is calculated as follows: let v_i and v_j denote the endpoints of e, and v_k and v_ℓ the other two vertices of the adjacent faces, as shown in Figure 4. The value of $\beta(e)$ is then defined to be the external angle of intersection between the circumcircles of the two triangles $v_j v_i v_k$ and $v_i v_j v_\ell$.

The total discrete Willmore energy of the mesh is then given by $W = \sum_{v \in V} W(v)$.

Bobenko also derives some properties of this energy, such as $W(v) \geq 0$ and $W(v) = 0$ if and only if v is convex and v and all its neighbours lie on a common sphere (possibly with infinite radius, i.e. a plane). It is therefore expected that a surface with minimum Willmore energy would be smooth and visually pleasing.

In order to use our algorithm to minimise this discrete analogue of Willmore energy we need to find the

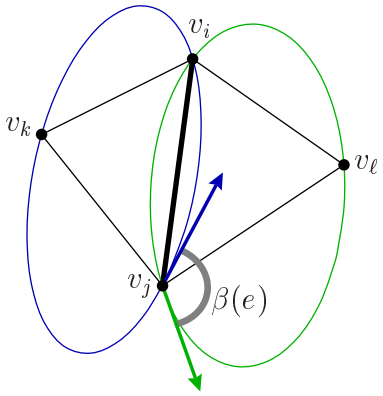

Figure 4: The angle $\beta(e)$ of an edge $e = (v_i, v_j)$ used in Bobenko's discrete analogue of Willmore energy.

corresponding reduction values, which we denote by $f_w(e)$, for every edge e.

Consider the four vertices v_i, v_j, v_k and v_ℓ as shown in Figure 4 and let $\beta_{ijk\ell}$ denote the angle $\beta(e)$ for the edge $e = (v_i, v_j)$ with adjacent vertices v_k and v_ℓ. Then the following two identities hold:

$$\beta_{ijk\ell} = \beta_{k\ell ij} \quad \text{and} \quad \beta_{ijk\ell} = \beta_{jik\ell}. \tag{3}$$

Refering to the labelling in Figure 3 and by using the identities (3), we have the following expression for $f_w(e)$, where e is the edge (v_i, v_j):

$$\begin{aligned} \tfrac{1}{2} f_w(e) \;=\; & \beta_{iktj} + \beta_{i\ell ju} + \beta_{jkiv} + \beta_{j\ell wi} \\ & -\beta_{ikt\ell} - \beta_{i\ell ku} - \beta_{jk\ell v} - \beta_{j\ell wk}. \end{aligned} \tag{4}$$

Again, the reduction values of certain edges should be updated after an edge flip. Since the reduction value of the edge (v_i, v_j) is dependent on the 8 vertices v_i, v_j, v_k, v_ℓ, v_t, v_u, v_v and v_w, the reduction values of the following 13 edges should be updated: (v_i, v_j), (v_i, v_k), (v_i, v_ℓ), (v_j, v_k), (v_j, v_ℓ), (v_i, v_u), (v_u, v_ℓ), (v_ℓ, v_w), (v_w, v_j), (v_j, v_v), (v_v, v_k), (v_k, v_t) and (v_t, v_i). Some of these edges may of course be the same depending on the mesh. The reduction values of all the other edges are unaffected by the flip.

Section 4 shows some results of applying our algorithm with this Willmore energy criterion on a few test models.

3.3 Localised Willmore energy

The next and final cost function defined in this paper is a modification of the Willmore energy criterion discussed in the previous section. This criterion was originally developed in an attempt to localise Willmore energy to a single edge. As illustrated in section 4 this

localisation could quite possibly provide a link between the two criteria discussed above in sections 3.1 and 3.2.

Consider an edge $e = (v_i, v_j)$ with adjacent faces $v_i v_k v_j$ and $v_i v_j v_\ell$. In the definition of our localised Willmore energy we temporarily add a vertex v_m to the mesh by positioning it at $\frac{1}{2}(v_i + v_j)$, i.e. the center of edge e, and connect it to v_i, v_j, v_k and v_ℓ as illustrated in Figure 5. We also add a temporary point v_n to the mesh after the flipping of e, positioned at $\frac{1}{2}(v_k + v_\ell)$, i.e. the center of the flipped edge, and connect it as shown.

The Willmore energy of v_m is then compared to the Willmore energy of v_n to determine the reduction value of e, which we denote by $f_l(e)$. Heuristically, the idea is that this would reflect what happens to the actual Willmore energy on a more local scale.

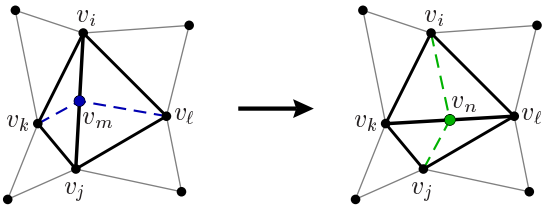

Figure 5: How the localised Willmore energy is defined for an edge $e = (v_i, v_j)$. Vertices v_m and v_n are temporarily added to the meshes before and after e is flipped.

Using the labelling from Figure 5 the reduction value for the edge e is expressed as

$$\begin{aligned} f_l(e) \;=\; & \beta_{mi\ell k} + \beta_{mjk\ell} + \beta_{mkij} + \beta_{m\ell ji} \\ & -\beta_{ni\ell k} - \beta_{njk\ell} - \beta_{nkij} - \beta_{n\ell ji}, \end{aligned} \qquad (5)$$

with v_m the temporary vertex in the center of edge (v_i, v_j) (before flipping) and v_n the temporary vertex in the center of edge (v_k, v_ℓ) (after flipping).

Again, since these reduction values depend on the 8 vertices v_i, v_j, v_k, v_ℓ, v_t, v_u, v_v and v_w, the same 13 edges need to be updated as those from the Willmore energy criterion in section 3.2.

We now have three different criteria (area, Willmore energy and localised Willmore energy) for the mesh optimisation algorithm outlined in section 2. In the following section some experimental results and a comparative discussion of these criteria are given.

4 Results and discussion

The algorithm described in section 2 was implemented with the different cost functions from section 3 and applied to a number of test models.

The first model is a triangulation fitted to points sampled from the surface of a torus. Figure 6 shows the initial mesh on the top left. For every criteria (area, Willmore energy and localised Willmore energy) we obtain the exact same mesh with our optimisation algorithm, shown on the top right. The figure also shows Gaussian curvature plots of these meshes subdivided by Loop's scheme [7]. An improvement in visual smoothness is evident.

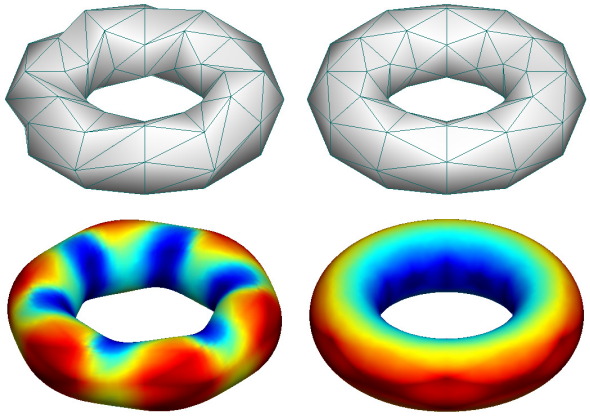

Figure 6: Example 1 (torus): initial mesh top left and optimised mesh top right. The bottom row shows Gaussian curvature plots of these two meshes subdivided with Loop's scheme. There is an obvious improvement in visual smoothness.

The next test model was obtained from a facial capturing system. Results are given in Figure 7. The intial mesh is shown followed by the three results obtained from minimising area, Willmore energy and localised Willmore energy. Below each mesh is a Gouraud shaded close-up of the top parts. The figure also shows plots of r vs area and r vs Willmore energy, indicating the change in area and Willmore energy as the algorithm with the different criteria proceed.

The initial mesh contains many "vertical" edges which are parameterisation artifacts. This explains the many vertical creases visible in the shaded version of the initial surface. Our optimisation algorithm minimising either area or localised Willmore energy seems to smooth out these creases which is desirable. When minimising Willmore energy most of these creases disappear although some are still visible.

The area and Willmore energy plots display some interesting behaviour. It seems that when the localised Willmore energy is minimised then surface area is also minimised. Also, when area or localised Willmore energy is minimised the Willmore energy seems to be minimised. In this case the algorithm actually reaches

a better local minimum in the Willmore energy by minimising either area of localised Willmore energy.

The third test model is a simplified version of the Stanford bunny. Figure 8 shows the initial mesh and the meshes obtained by minimising area, Willmore energy and localised Willmore energy. Plots of r vs area and r vs Willmore energy are also shown.

This example illustrates a problem that might typically arise when minimising area (or localised Willmore energy apparently). Narrow tube-like areas of the mesh such as the ears of the bunny suffer quite visibly from a reduction in volume. This is usually an undesired effect.

Once again a strong relationship between minimising area and minimising localised Willmore energy is evident from the plots in Figure 8. Also, even though minimising these two functions may at some stages cause the Willmore energy to increase slightly, the overall effect seems to be minimising.

All of the examples above suggest some relationship between minimising area and minimising localised Willmore energy. Our experimental results led us to the following conjecture (a proof for or against is yet to be found):

Conjecture : *For any edge e of a mesh we have $f_a(e) \leq 0$ if and only if $f_l(e) \leq 0$, with f_a and f_l given by (1) and (5).*

If the statement above is true it would explain why the curves corresponding to minimising localised Willmore energy in the area plots of Figures 7 and 8 are both monotonically decreasing.

Also, if we can determine exactly why it seems that the localised Willmore energy scheme also minimises the actual Willmore energy (as is suggested in the Willmore energy plots of Figures 7 and 8) then our conjecture may lead to a concrete relationship between the minimisation of Willmore energy and the minimisation of area.

5 Conclusion

We presented a general framework for a mesh optimisation algorithm. This algorithm is designed to minimise a given cost function by successively flipping edges of some initial input mesh. The algorithm may terminate in a local minimum of the cost function, and techniques of escaping such situations are topics for future research.

We also defined three different cost functions that the algorithm can attempt to minimise based on area, Willmore energy and a localised Willmore energy scheme. The last of these introduces temporary vertices corresponding to each edge in the mesh.

Results from applying our algorithm on a few test models suggest evidence of a strong relationship between the minimisation of area and the minimisation of localised Willmore energy. We conjecture that an edge flip decreases the total surface area if and only if the localised Willmore energy decreases. If proven true this may lead to a relationship between the Willmore energy of a surface and its area.

References

[1] L. Alboul, G. Kloosterman, C. Traas and R. van Damme. *Best data-dependent triangulations.* Journal of computational and applied mathematics, 119:1–12, 2000.

[2] F. Bernardini and C.L. Bajaj. *Sampling and reconstructing manifolds using alpha-shapes.* Proceedings of the 9th Canadian conference of computational geometry, 193–198, 1997.

[3] A.I. Bobenko. *A conformal energy for simplicial surfaces.* Combinatorial and computational geometry, MSRI Publications, 52:133–143, 2005.

[4] A.I. Bobenko and P. Schröder. *Discrete Willmore flow.* Eurographics symposium on geometry processing, 101–110, 2005.

[5] N. Dyn, K. Hormann, S.J. Kim and D. Levin. *Optimizing 3D triangulations using discrete curvature anlysis.* Mathematical methods for curves and surfaces, Oslo, 135–146, 2000.

[6] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald and W. Stuetzle. *Surface reconstruction from unorganized points.* ACM SIGGRAPH '92, 71–78, 1992.

[7] C. Loop. *Smooth subdivision surfaces based on triangles.* Master's Thesis, Utah University, USA, 1987.

[8] J. O'Rurke. *Triangulation of minimal area as 3D object models.* Proceedings of the international joint conference on AI 81, Vancouver, 664–666, 1981.

[9] T.J. Willmore. *Riemannian geometry.* Clarendon Press, Oxford, 1993.

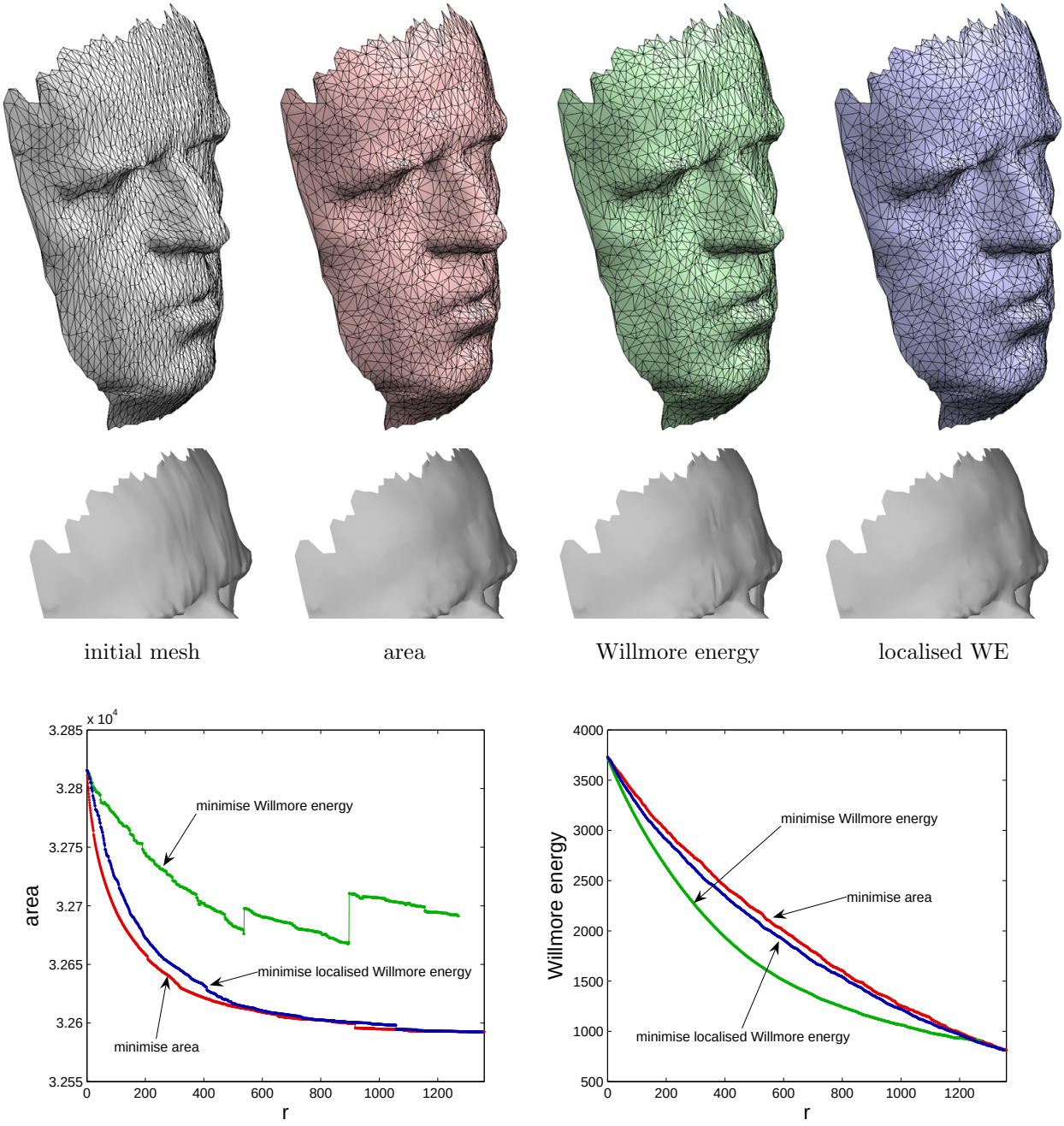

Figure 7: Example 2 (face): The initial mesh and meshes resulting from minimising area, Willmore energy and localised Willmore energy are shown on the top row. The second row depicts the top parts of the same meshes with gouraud shading giving an indication of how smooth each is. On the bottom row plots of r vs area and r vs Willmore energy are shown for each of the three criteria.

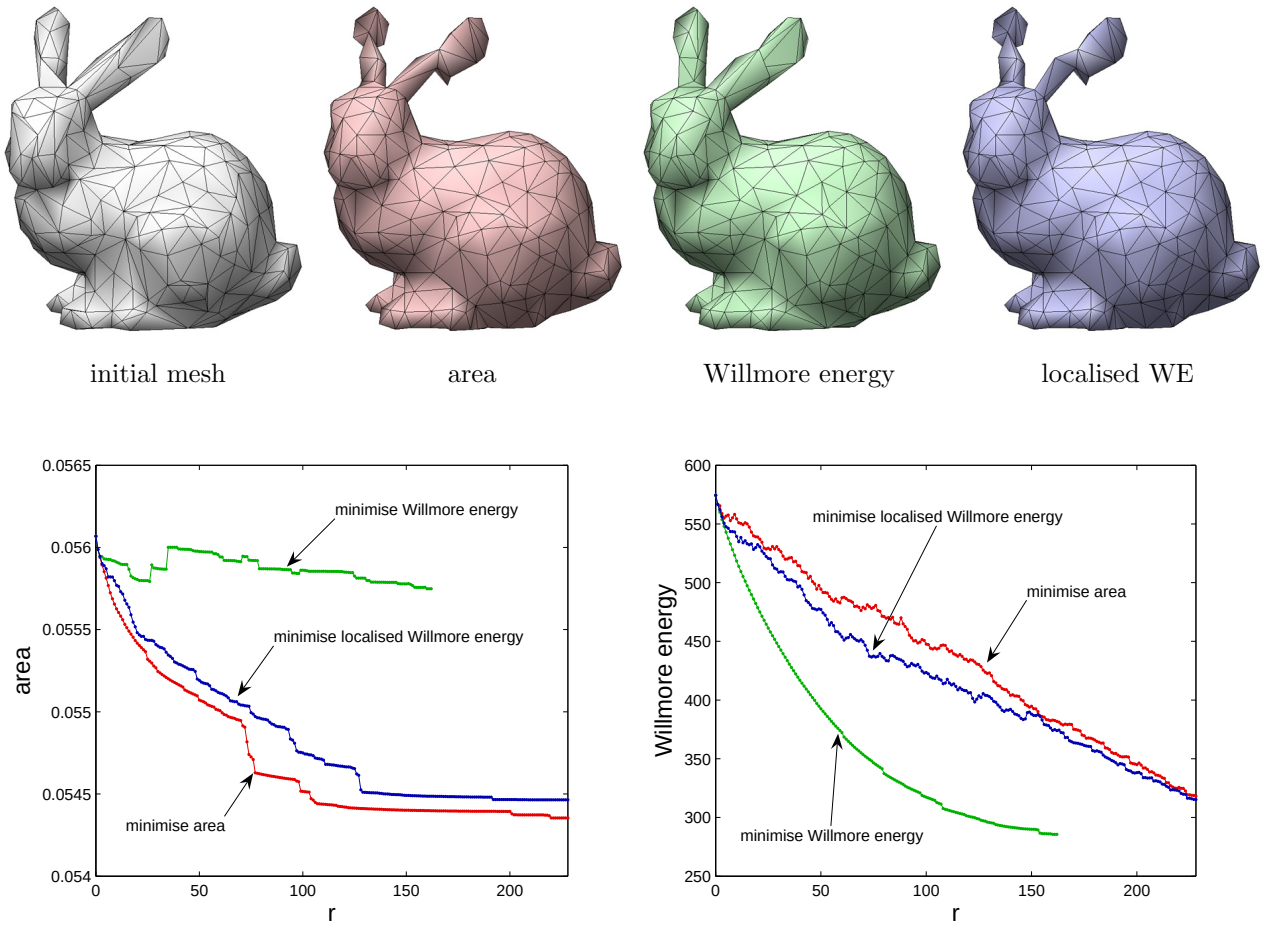

Figure 8: Example 3 (bunny): The initial mesh and meshes resulting from minimising area, Willmore energy and localised Willmore energy are shown on the top row. On the bottom row plots of r vs area and r vs Willmore energy are shown for each of the three criteria.