

Appendix of Computing a High-Dimensional Euclidean Embedding from an Arbitrary Smooth Riemannian Metric

A DERIVATION OF EQ. (16) AND EQ. (17)

In this appendix we show the detailed derivation of Eq. (16) and Eq. (17), which is inspired by the idea given in the Appendix of Shoemake and Duff’s paper [Shoemake and Duff 1992].

We only consider E_{em} in this case, since E_{reg} is constant when the vertex coordinates $\{\bar{v}_i | i = 1, \dots, n_v\}$ are fixed. Note that for E_{em} in surface case, i.e., Eq. (12), $T_j = \bar{W}_j P_j^{-1} O_{ja}^T$ or in volume case, i.e., Eq. (13), $T_j = \bar{W}_j W_j^{-1}$ is also constant when $\{\bar{v}_i | i = 1, \dots, n_v\}$ are fixed. The high-dimensional rotation \bar{U}_j for each triangle or tetrahedron j is independent of each other, so our problem is simply expressed as:

For triangle or tetrahedron j , find \bar{U}_j minimizing $\|T_j - \bar{U}_j Q_j\|_F^2$, subject to orthogonality constraint $\bar{U}_j^T \bar{U}_j - I = 0$.

Since the squared Frobenius norm $\|T_j - \bar{U}_j Q_j\|_F^2$ is exactly the trace of matrix $(T_j - \bar{U}_j Q_j)(T_j - \bar{U}_j Q_j)^T$, we can use a symmetric Lagrange multiplier matrix Y to incorporate the orthogonality constraint as a linear term in the Lagrangian:

$$F(\bar{U}_j, Y) = \text{trace}[(T_j - \bar{U}_j Q_j)(T_j - \bar{U}_j Q_j)^T + (\bar{U}_j^T \bar{U}_j - I)Y]. \quad (1)$$

By differentiating $F(\bar{U}_j, Y)$ with respect to \bar{U}_j and making it equal to a zero matrix, we can get:

$$2(\bar{U}_j Q_j - T_j)Q_j^T + 2\bar{U}_j Y = 0. \quad (2)$$

Thus we have:

$$\bar{U}_j(Q_j Q_j^T + Y) = T_j Q_j^T. \quad (3)$$

Since $Q_j Q_j^T + Y$ is a symmetric matrix and \bar{U}_j is an orthogonal matrix, we can consider $\bar{U}_j(Q_j Q_j^T + Y)$ as the Polar decomposition of matrix $T_j Q_j^T$.

This means that \bar{U}_j is the orthogonal factor of the Polar decomposition of matrix $T_j Q_j^T$, as shown in Eq. (16) and Eq. (17) in paper.

B BEHAVIORS OF DIFFERENT μ VALUES IN EQ. (15)

The coefficient μ is a weighting factor to balance the similarity and regularity terms during embedding optimization (Sec. 3.4.1 in paper). The similarity term is defined based on the number of mesh elements (i.e., triangles or tetrahedrons), and the regularity term is defined based on the number of vertices. Through Euler characteristic, we know that the number of mesh elements and the number of vertices in a manifold have the linear relationship. So no matter what mesh models are, it is possible to find a suitable constant coefficient μ to balance the similarity and regularity energies at a relatively stable order of magnitude in different models. From our experimental

observations, $\mu = 100$ (i.e., order of magnitude of μ is 2) is a good choice to all models in our paper and appendix.

In this section, we provide the experiments for behaviors of different μ values in embedding results. In order to emphasize its importance and behaviors, we choose different orders of magnitude of values to test the corresponding embedding results. Since it is not straightforward to visualize the computed embedding results $\geq 4D$, we choose the example of an ideal embedding in 3D space, i.e., 2D domain with Gaussian embedding metric (Fig. 4 in paper) as examples to illustrate the importance of having an appropriate value of the coefficient in the regularity term.

If we use Eq. (12) or Eq. (13) only, i.e., without adding the regularity term E_{reg} to optimize a 3D embedding, the result is shown in Fig. 1 ($\mu = 0$). We can see that the result is far from our desired result (many wrinkles and not smooth), known as corrugations, that can form fractal patterns [Borelli et al. 2012], though the average relative edge length error does reduce dramatically (i.e., $L_{avg}^{rel} = 11.73\%$ compared with original $L_{avg}^{rel} = 20.71\%$). So it is necessary to add a regularity term to obtain a smooth embedding.

Along with increasing the orders of magnitude of μ , we can see that the embedding is smoother. It is interesting to see that the embedding errors are decreasing at first and then increasing, this is probably because some embedding results are at local minima (such as $\mu = 0, 1, 10$ in Fig. 1), so adding a regularity term is also a good strategy to let the optimizer jump out of the local minima to reach a better result. However, if the regularity coefficient is too large, the embedding is over-smooth and the accuracy of the embedding is poor. One extreme case is the order of magnitude of μ is 5 (i.e., $\mu = 100,000$), the computed embedding almost does not have any deformation from the original 2D plane. In order to balance the smoothness and accuracy, we have tried a sequence of increasing values of μ in different orders of magnitude, and found that $\mu = 100$ is an appropriate setting ($L_{avg}^{rel} = 0.92\%$ and $L_{max}^{rel} = 10.58\%$; and the 3D embedding shape is smooth and very close to the ideal Gaussian surface) as shown in Fig. 1.

C OBSERVATION ON CHOOSING THE DIMENSION OF EMBEDDINGS

Tab. 1 provides the average relative edge length errors of all surface and volume models in our experiments (besides the results shown in Sec. 6.1.2 of paper) for different dimensions. Error values in bold demonstrate that 8D is a good and reasonable choice observed in the experiments.

D K-NN EFFICIENCY ON HIGH-D EMBEDDING

The following criteria of K-NN efficiency are used: T_{knn} is computational time on K-NN searching for all particles; N_{avg} is the average number of nearest neighbors to be searched; N_{max} is the maximal number of nearest neighbors to be searched. We use the

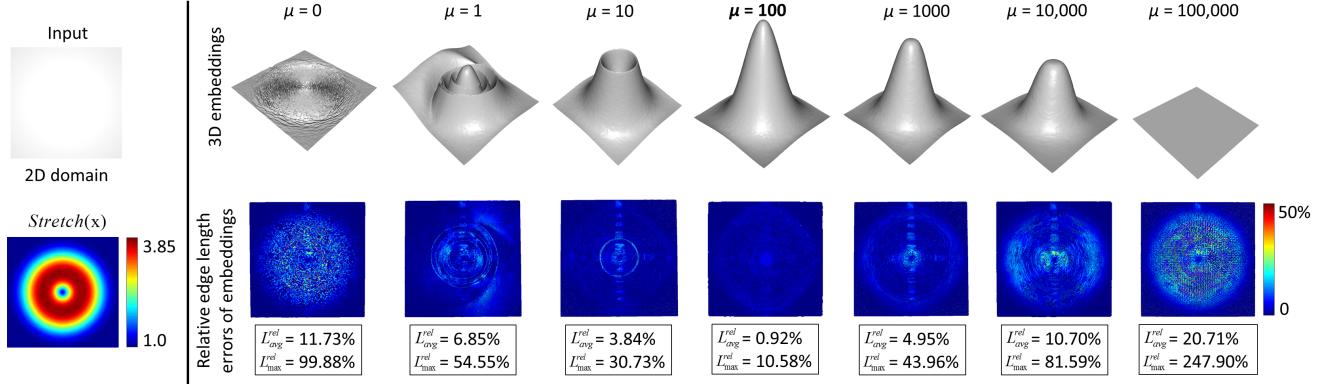


Fig. 1. The behaviors of different orders of magnitude of μ values in 3D embedding results of a 2D domain with Gaussian embedding metric. A μ value in bold emphasizes the best result observed in the experiment.

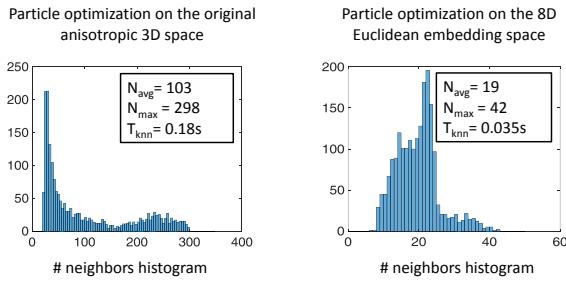


Fig. 2. Comparison of K-NN searching of 2000 particles optimization on the original anisotropic 3D Cyclide surface and its 8D Euclidean embedded surface.

above criteria to compare Zhong et al.’s anisotropic particle optimization [2013] with our proposed isotropic particle optimization on the high-d Euclidean embedding.

In order to compare the efficiency of the K-NN between the high-d embedding space and the original space, we use the same search radius 5σ , where σ is the kernel width of the inter-particle energy in the Euclidean embedding space (as mentioned in Sec. 4.1 of paper). However, when the K-NN computation happens in the original space, we need to transform this search radius into the original anisotropic space, resulting in a large range once the anisotropic stretching ratio is high. After that, we need to further check and prune the spurious neighbors under the given metric.

Fig. 2 shows the K-NN computation of 2000 particles optimization on the computed 8D Euclidean embedding in this paper and the original 3D surface [Zhong et al. 2013] of the Cyclide model with the stretching ratio $\frac{s_2}{s_1} \in [1.6, 9.4]$. We provide the statistics when the particles are at equilibrium state, i.e., at the end of the optimization, since the number of neighbors may vary slightly at each iteration during optimization. The K-NN at equilibrium state is more convincing and stable to analyze. It demonstrates that the average K-NN on 8D Euclidean embedding is 19 compared with 103 on the original anisotropic space, and the K-NN searching time of 2000 particles for each iteration on the embedded surface is about 5 times faster than that of the original surface with the specified anisotropic metric.

Besides Cyclide model, we also measure the K-NN efficiency on the high-d embeddings of other models, such as Kitten, Vase, Knot, Club, etc., and the average K-NN is quite stable and small, i.e., $N_{avg} = 20$. However, without using Euclidean embedding, the average K-NN highly depends on the stretching ratio in the Riemannian metric. The higher the stretching ratio is, the larger the average K-NN is, such as $N_{avg} = 232$ on another Cyclide model (as shown in Fig. 13 of paper) with the larger stretching ratio $\frac{s_2}{s_1} \in [2, 29]$.

E MORE RVD AND MESHING RESULTS

Our proposed method works on different shapes of the models and there are some extra results given in the following, including different topology genera, closed surfaces and surfaces with boundaries. Figs. 3, 4, 5, 6, 7, 8, and 9 show that the anisotropic 3D surface RVD and meshing results of Vase, Bone, Knot, Genus3, Rocker Arm, Club, and Hand models with the anisotropic metrics designed by the surfaces’ curvature tensors. In order to demonstrate the scalability of the proposed method on medium-large sizing RVD and meshing of 3D surfaces, Fig. 10 (Fertility model) shows the final results with vertices ranging from 10,000 to 100,000. The statistics and timings for our 8D embedding computation and surface meshing on these models are shown in Tab. 2. It is noted that our embedding computation in 8D space is quite efficient, and all surface models provided in this article need only dozens of seconds (only the large sizing model, such as the dense Fertility model, needs hundreds of seconds).

Fig. 11 show some more volumetric RVD results on a Cube model with domain $[1, 11]^3$ and the sampling ranges from 6000 to 30,000. The targeted Riemannian metric field is specified via a highly nonlinear analytic function as $M(x) = R(x)^T \text{diag}(\text{Stretch}(x)^2, 1, 1)R(x)$, where $\text{Stretch}(x) = (0.025 + (1 - e^{-0.01|x^2+y^2+z^2-49|}))^{-1}$, and $R(x)$ ’s columns are $(x/\sqrt{x^2+y^2+z^2}, y/\sqrt{x^2+y^2+z^2}, z/\sqrt{x^2+y^2+z^2})^T$, and two orthogonal vectors. The stretching ratio is $\text{Stretch}(x) \in [1, 40]$. Fig. 12 shows the visualization of the stress tensor field by using 3D volumetric RVD (with 10,000 samples) on the Brake Lever model (filled with air inner part) with the stretching factor in the major direction $s_1(x) \in [0.5, 103]$ (data downloaded from www.tensorvis.org).

F MORE COMPARISON RESULTS

In order to further demonstrate the better performance of our method compared with other anisotropic meshing approaches, we provide more experiments in the following, in addition to results in the paper. All the meshing results of the comparison methods were provided by the original authors.

Fig. 13 compares our method with anisotropic Delaunay refinement (ADR) [Boissonnat et al. 2015] on the Fertility surface model with the stretching ratio $\frac{s_2}{s_1} \in [1, 14]$.

Fig. 14 compares our method with conformal embedding [Zhong et al. 2014] and particle-based method [Zhong et al. 2013] on the Ellipsoid surface model with the stretching ratio $\frac{s_2}{s_1} \in [1, 10]$.

Tab. 3 shows the final mesh quality of different anisotropic surface methods. Our results demonstrate that they can yield better mesh angle and triangle quality based on the proposed high-d embedding framework, so that match the input curvature-based anisotropy.

REFERENCES

- J-D. Boissonnat, K. Shi, J. Tournais, and M. Yvinec. 2015. Anisotropic Delaunay Meshes of Surfaces. *ACM Trans. Graph.* 34, 2 (2015), 14.
- V. Borelli, S. Jabrane, F. Lazarus, and B. Thibert. 2012. Flat Tori in Three-dimensional Space and Convex Integration. *Proceedings of the National Academy of Sciences of the United States of America* 109, 19 (2012).
- K. Shoemake and T. Duff. 1992. Matrix Animation and Polar Decomposition. In *Proceedings of the Conference on Graphics Interface*. 258–264.
- Z. Zhong, X. Guo, W. Wang, B. Lévy, F. Sun, Y. Liu, and W. Mao. 2013. Particle-Based Anisotropic Surface Meshing. *ACM Trans. Graph.* 32, 4 (2013), 99:1–99:14.
- Z. Zhong, L. Shuai, M. Jin, and X. Guo. 2014. Anisotropic Surface Meshing with Conformal Embedding. *Graphical Models* 76, 5 (2014), 468–483.

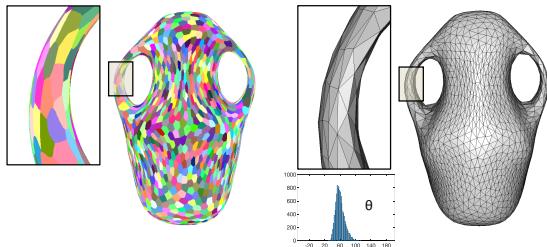


Fig. 3. Vase model.

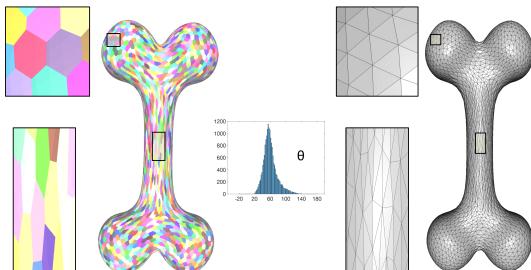


Fig. 4. Bone model.

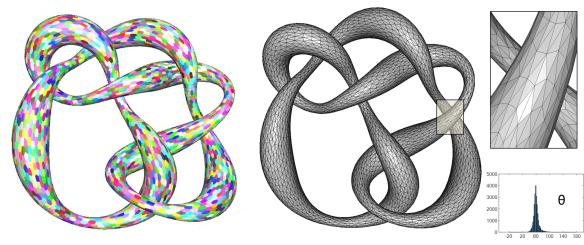


Fig. 5. Knot model.

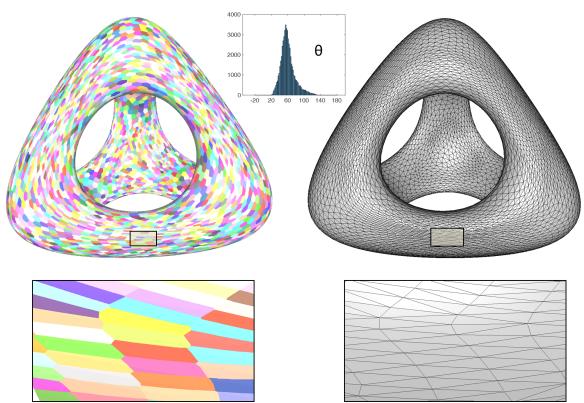


Fig. 6. Genus3 model.

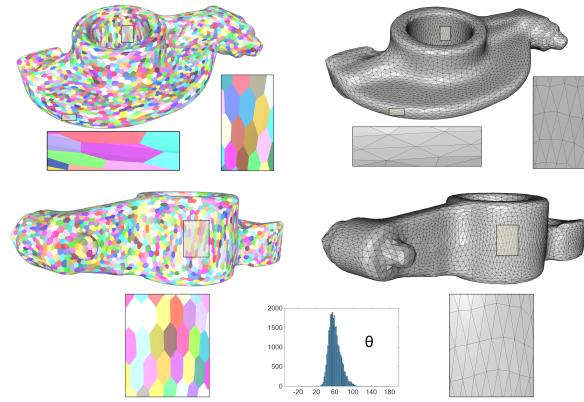


Fig. 7. Rocker Arm model.

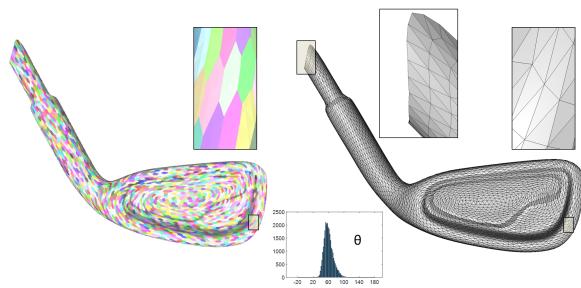


Fig. 8. Club model with boundaries.

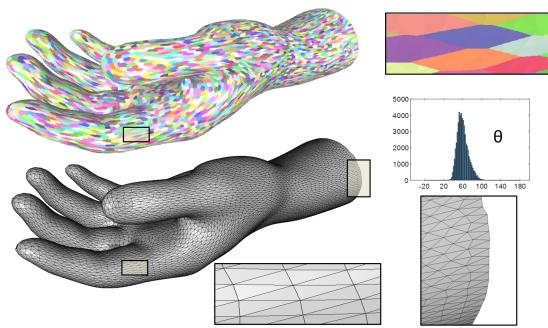


Fig. 9. Hand model with boundaries.

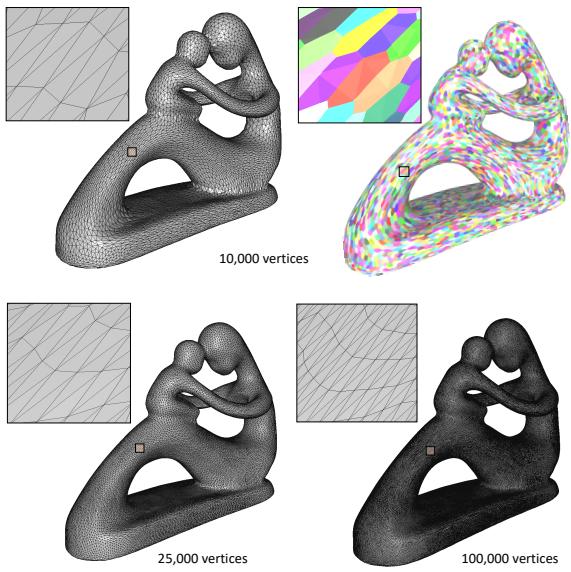


Fig. 10. Fertility models with 10,000, 25,000, and 100,000 vertices.

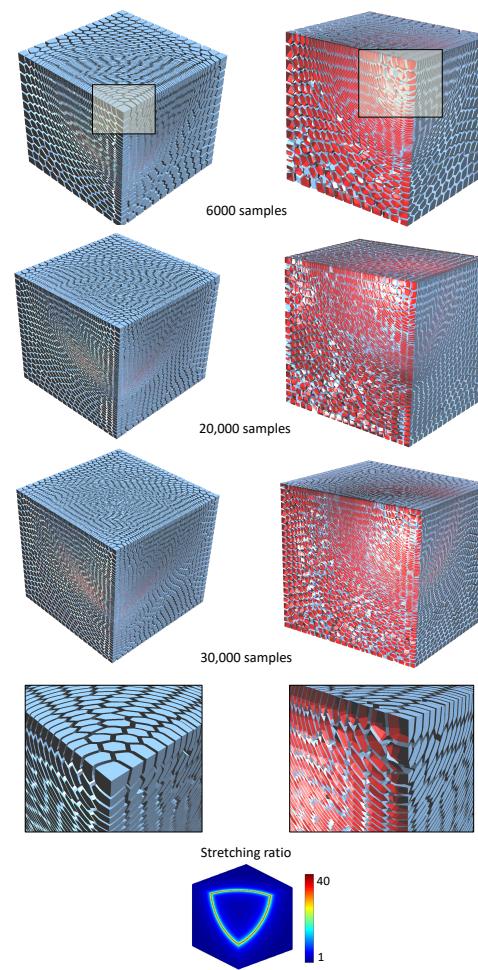
Fig. 11. The anisotropic 3D volumetric RVD results on the Cube model with 6000, 20,000, and 30,000 samples. The stretching ratio is $\text{Stretch}(x) \in [1, 40]$.

Table 1. Statistics for embedding errors (i.e., average relative edge length errors) in different dimensions.

	Model	4D	5D	6D	7D	8D	10D	20D
Surfaces	Gargo	16.70%	10.07%	7.38%	6.90%	6.72%	6.62%	6.53%
	Upright	22.23%	4.69%	3.00%	2.65%	1.52%	1.48%	1.46%
	Nefertiti	14.32%	6.39%	3.78%	3.24%	2.98%	2.95%	2.93%
	Vase	15.71%	8.69%	5.12%	4.30%	4.18%	4.15%	4.10%
	Bone	29.41%	12.09%	6.33%	4.15%	3.74%	3.55%	3.40%
	Knot	20.22%	8.70%	6.11%	4.86%	4.49%	4.38%	4.31%
	Genus3	17.46%	7.75%	4.37%	3.91%	3.70%	3.59%	3.46%
	Rocker Arm	16.19%	10.90%	6.08%	5.77%	5.62%	5.59%	5.59%
	Club	17.09%	10.04%	7.26%	6.18%	5.44%	5.40%	5.35%
	Hand	14.69%	5.81%	3.06%	2.88%	2.68%	2.62%	2.47%
Volumes	Fertility	20.60%	9.35%	5.68%	4.09%	3.86%	3.71%	3.68%
	Ellipsoid	35.69%	5.14%	2.60%	1.65%	0.96%	0.94%	0.92%
	Cube2 (Fig. 15 b in paper)	13.20%	4.48%	2.90%	1.56%	1.30%	1.22%	1.17%
	Cube3 (Fig. 16 in paper)	19.34%	7.78%	4.87%	2.75%	2.12%	2.10%	2.07%
	Sphere (Fig. 18 in paper)	32.14%	13.13%	4.34%	2.18%	2.04%	2.02%	2.01%

Note: The values in bold are the 8D embedding results.

Table 2. Statistics and timings for 8D embedding computation and surface meshing on models in appendix.

Model	Input #Vert.	Stretch	L_{avg}^{rel}	L_{max}^{rel}	Embed. Time	Output #Vert.	G_{min}	G_{avg}	θ_{min}	θ_{avg}	$\%_{<30^\circ}$	Mesh Time
Vase	20, 000	[1, 5]	4.18%	157.97%	17.09 s	2000	0.36	0.86	21.33°	49.52°	0.24%	7.48 s
Bone	16, 794	[1, 10]	3.74%	125.24%	13.45 s	3000	0.32	0.85	19.69°	48.47°	0.48%	8.69 s
Knot	24, 392	[2, 8]	4.49%	168.46%	21.16 s	5000	0.47	0.93	25.37°	53.79°	0.05%	26.41 s
Genus3	26, 620	[1, 10]	3.70%	110.08%	40.48 s	8000	0.42	0.90	23.12°	50.51°	0.09%	35.16 s
Rocker Arm	35, 840	[1, 7]	5.62%	138.31%	32.38 s	5000	0.33	0.86	19.54°	48.13°	0.15%	31.47 s
Club	30, 000	[1, 6]	5.44%	143.44%	27.82 s	5000	0.46	0.91	24.10°	51.25°	0.07%	29.94 s
Hand	36, 619	[1, 5]	2.68%	121.57%	35.78 s	10, 000	0.28	0.90	16.28°	50.14°	0.23%	55.50 s
Fertility	55, 902	[1, 10]	3.86%	169.84%	63.31 s	10, 000	0.32	0.86	18.02°	48.10°	0.21%	51.53 s
Fertility	55, 902	[1, 10]	3.86%	169.84%	63.31 s	25, 000	0.46	0.90	25.16°	51.51°	0.05%	121.18 s
Fertility	223, 626	[1, 10]	2.58%	97.36%	256.44 s	100, 000	0.62	0.92	28.00°	53.55°	0.006%	456.39 s

Note: Embed. Time: timing for embedding computation with 50 iterations. Mesh Time: timing for both particle optimization with 50 iterations and RVD computation.

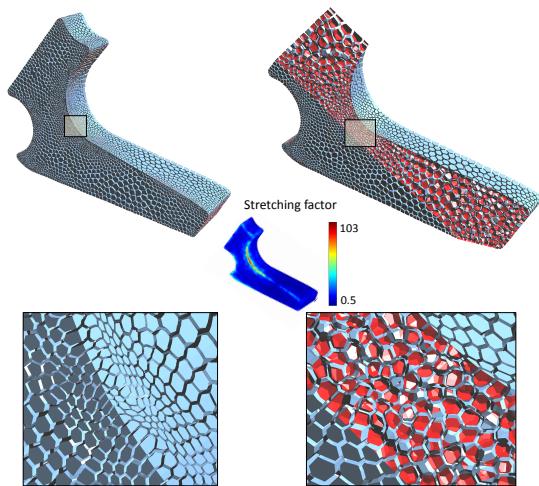


Fig. 12. The visualization of the stress tensor field by using 3D volumetric RVD (with 10,000 samples) on the Brake Lever model (filled with air inner part) with the stretching factor in the major direction $s_1(x) \in [0.5, 103]$.

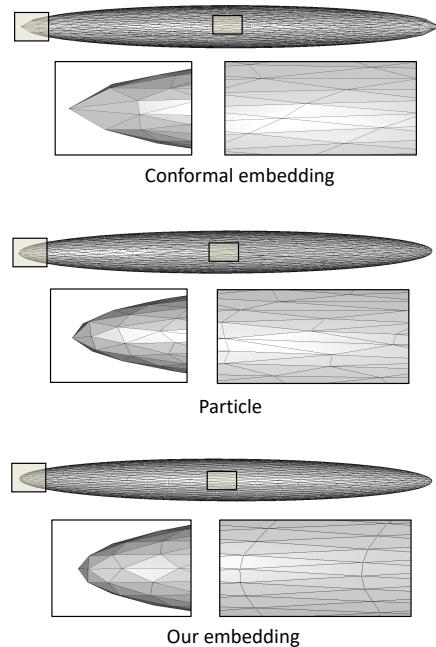


Fig. 14. Comparison on anisotropic meshing results (1000 vertices) with conformal embedding method [Zhong et al. 2014], particle-based method [Zhong et al. 2013], and our method on the Ellipsoid surface model.

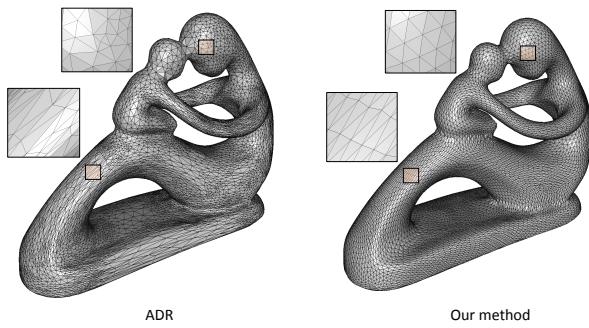


Fig. 13. Comparison on anisotropic meshing results (12,480 vertices) with ADR [Boissonnat et al. 2015] and our method on the Fertility surface model.

Table 3. Comparison of mesh quality on anisotropic surface meshing methods.

Model	Stretch	Output # Vert.	G_{min}	G_{avg}	θ_{min}	θ_{avg}	$\%_{<30^\circ}$
Ellipsoid (conformal embed)	[1, 10]	1000	0.11	0.78	3.93°	44.03°	12.42%
Ellipsoid (particle)	[1, 10]	1000	0.45	0.84	18.96°	48.50°	0.15%
Ellipsoid (our high-d embed)	[1, 10]	1000	0.57	0.94	36.57°	55.13°	0
Fertility (ADR)	[1, 14]	12, 480	0.002	0.56	0.06°	29.94°	41.79%
Fertility (our high-d embed)	[1, 14]	12, 480	0.39	0.90	19.02°	51.17°	0.17%

Note: The values in bold emphasize the best results observed in the comparison experiments.