

Supplementary for Discretization-Agnostic Deep Self-Supervised 3D Surface Parameterization

Chandradeep Pokhariya *

chandradeep.pokhariya@research.iiit.ac.in
IIIT Hyderabad
India

Asttva Srivastava

asttva.srivastava@research.iiit.ac.in
IIIT Hyderabad
India

CCS CONCEPTS

- Computing methodologies → Parametric curve and surface models; Neural networks.

KEYWORDS

UV parameterization, texture mapping, neural networks, multi-layer perceptron, self-supervised learning, computer graphics.

ACM Reference Format:

Chandradeep Pokhariya, Shanthika Naik, Asttva Srivastava, and Avinash Sharma. 2022. Supplementary for Discretization-Agnostic Deep Self-Supervised 3D Surface Parameterization. In *SIGGRAPH Asia 2022 Technical Communications (SA '22 Technical Communications)*, December 6–9, 2022, Daegu, Republic of Korea. ACM, New York, NY, USA, 4 pages. <https://doi.org/10.1145/3550340.3564235>

1 RELATED WORKS

Conventional Methods for Surface Parameterization: Conventional methods to solve mesh parameterization generally fall into one of the three categories. The first one is *single-patch, fixed boundary* methods, e.g. harmonic parameterization[Wang et al. 2013], which projects the boundary vertices onto a circle in UV space and computes two harmonic functions (one for u and one for v coordinate). LSCM[Lévy et al. 2002], which is a *single-patch, free boundary* parameterization method, minimizes the conformal (angular) distortion. Unlike harmonic parameterization, it does not need to have a fixed boundary. Both the aforementioned categories can only deal with open surfaces with genus 0. The third category is formally known as *global parameterization* method, which can deal with meshes of arbitrary genus. They achieve this by cutting the given mesh into the patch(es) and individually parameterizing each patch. The generated per-patch maps are discontinuous around the cut when laid down in the UV space. This discontinuity can be seen

*Both authors contributed equally to this research.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

SA '22 Technical Communications, December 6–9, 2022, Daegu, Republic of Korea
© 2022 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-9465-9/22/12...\$15.00
<https://doi.org/10.1145/3550340.3564235>

Shanthika Naik*

shanthika.naik@research.iiit.ac.in
IIIT Hyderabad
India

Avinash Sharma

asharma@iiit.ac.in
IIIT Hyderabad
India

as seams on the 3D surface. Another class of global methods try to detect one or more seams to cut the mesh to make it open and then parameterize it. OptCuts[Li et al. 2018] and Boundary-First Flattening[Sawhney and Crane 2017] fall into this category. There are global seamless parameterization methods as well, but they are out of the scope of this work.

Neural Methods for Surface Parameterization: Neural parameterization methods have gained popularity in the past few years due to their ability to address ill-posed problems as well as advancements in deep learning methodologies and hardware stack. AtlasNet[Groueix et al. 2018] was one of the first works along these lines, which tries to generalize on different classes of objects. However, its use case was directed more towards surface reconstruction than surface parameterization. Another method, DGP[Williams et al. 2018], builds upon AtlasNet and proposes an object-centric way of surface reconstruction by overfitting a neural network representing a local chart parameterization. Both methods use a fixed number of patches for the surface parameterization but require a different neural network for every patch, which is overkill and difficult to scale. A recent work, AUV-Net[Chen et al. 2022], takes a point cloud as input and learns parameterization of aligned surfaces (e.g., faces and humans in T-poses) using a cycle-loss and smoothness loss. However they require all the geometries of the same category and in the same orientation. Moreover, their patch estimation method is very naive and can not scale to an arbitrary number of patches. All the aforementioned learning-based methods sample points in the UV space and learn to map it to a 3D surface, thereby assuming the UV space itself, hence failing to produce a plausible UV map. Another very recent method [Aigerman et al. 2022] learns intrinsic mapping of arbitrary surfaces in a supervised fashion where a conventional method acts as the ground truth.

2 IMPLEMENTATION DETAILS

For PatchNet, we use DiffusionNet[Sharp et al. 2020] architecture with 4-blocks, channel width of 128 and 64 eigenbasis vectors for spectral acceleration. We use ReLU activations at intermediate layers and softmax function after the final output layer. The surface parameterization module uses an 8 layer MLP with 1.3×10^6 parameters for both forward and backward MLP with LeakyReLU activations in-between the layers and *tanh* at the final output layer. We use the PatchNet loss weights $\{\lambda_{cos}, \lambda_{geo}\} = [1.0, 1.0]$ and the parameterization loss weights $\{\lambda_1, \lambda_2, \lambda_3, \lambda_4\} = [1.0, 1.0, 0.001, 0.001]$.

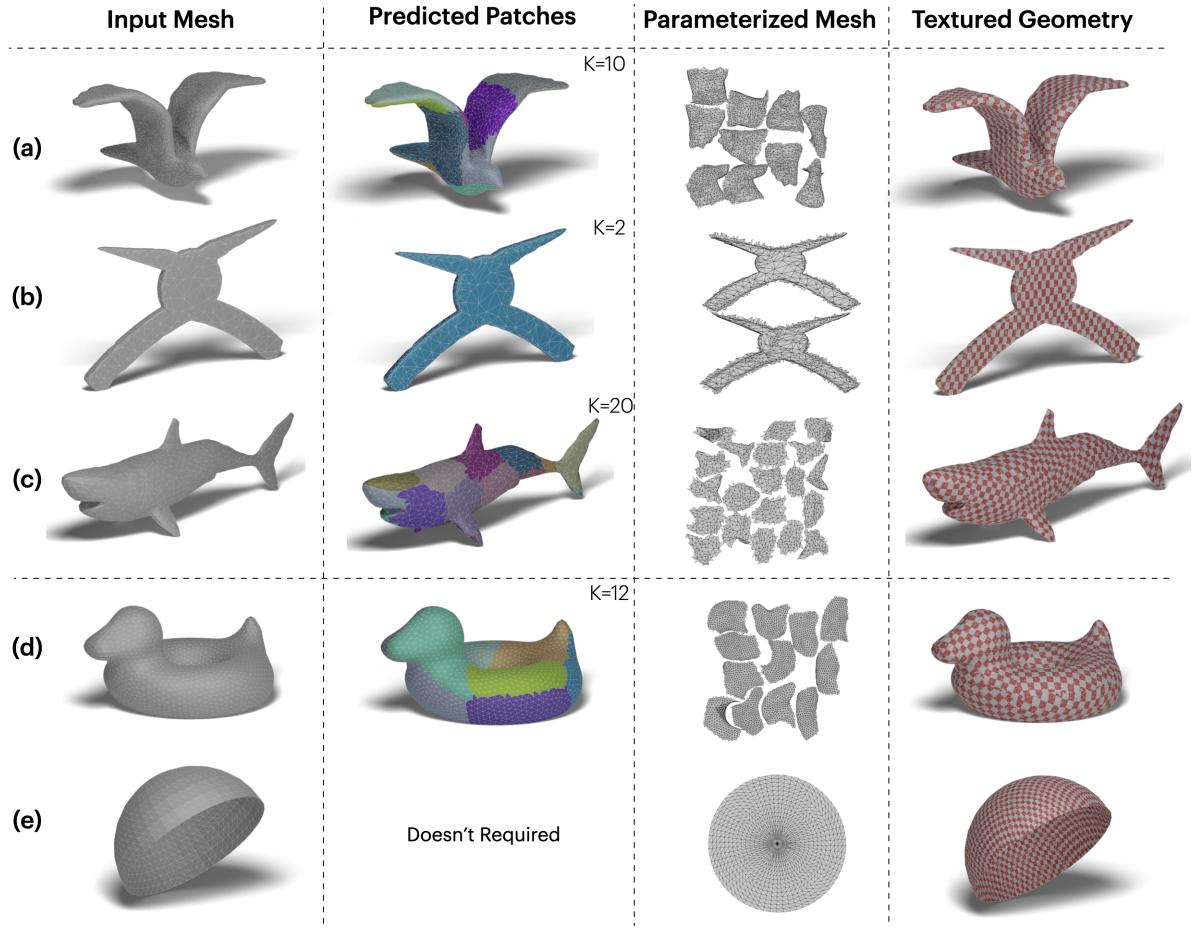


Figure 1: Additional Qualitative Results

We use ADAM optimizer with a learning rate of 10^{-3} and batch size of 1 on a single RTX 2080Ti GPU for all our experiments. The framework is implemented in PyTorch Lightning, trained on a single RTX 2080Ti GPU. We use xatlas[jpcy 2022] to pack the individual patches into the final UV atlas.

3 ADDITIONAL QUALITATIVE RESULTS

Figure 1 shows qualitative results of our framework on arbitrary closed ((a)-(d)) as well as open meshes. The patches are extracted and parameterized individually for open meshes to form a UV atlas. In the case of (e), our method estimates a reliable surface parameterization even with high extrinsic curvature. Moreover, meshes (a), (b) & (c) are the unseen test samples from their respective classes, which are directly inferred. On the other hand, for meshes (d) & (e), parameterization is obtained by training a network till convergence. This shows that apart from learning parameterization in an object-centric way, our framework can also generalize to a specific class/category and perform well on category-specific samples.

4 ABLATION STUDY

4.1 Patch Extraction Module

Effect of Geodesic Loss: In section-3.1, we stated that the geodesically far-apart faces with high-cosine similarity might get assigned to the same patch, producing unwanted patches of extreme curvature. However, incorporating \mathcal{L}_{geo} into the objective function tends to sort out this issue as it penalizes the faces geodesically



Figure 2: Effect of geodesic loss on the patch extraction module.

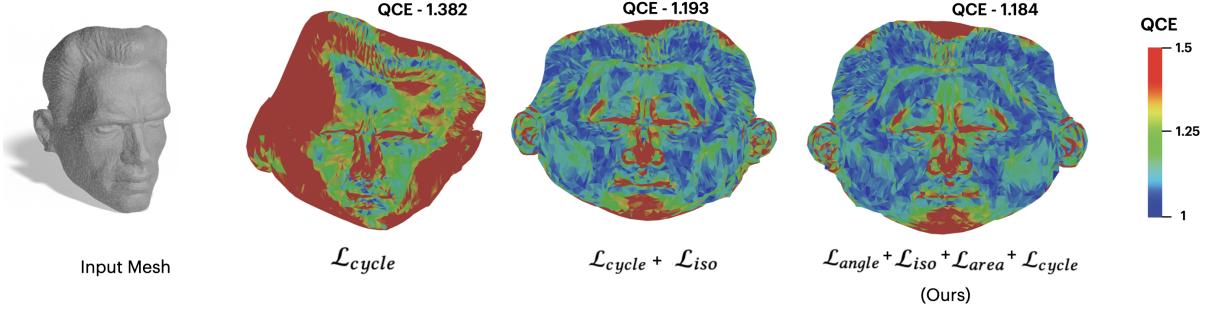


Figure 3: Ablation of different losses in surface parameterization module.

far apart and belonging to the same patch. This improvement is evident in Figure 2.

4.2 Surface Parameterization Module

Effect of DiffusionNet Embeddings: As described in the method section of the main draft, the MLP_f and MLP_{f-1} take a global encoding ψ as input along with vertex position. This global encoding/embedding is the combination of the DiffusionNet features of all the vertices. Instead of using global encoding, per-vertex features from DiffusionNet can also be passed directly to the MLPs as input. However, we argue that per-vertex features are noisy and capture minimal global context, resulting in an irregular UV space and undesired UV coordinates. It can be observed in Figure 4 that when global encoding of DiffusionNet features is incorporated, the quasi-conformal error (QCE) drops. Moreover, from Figure 5, it is clear that the global encoding ψ provides the better context of the global shape as compared to per-vertex embeddings, thereby regularizing the UV space and hence producing better output both qualitatively and quantitatively (minimal overlapping and lower QCE value).

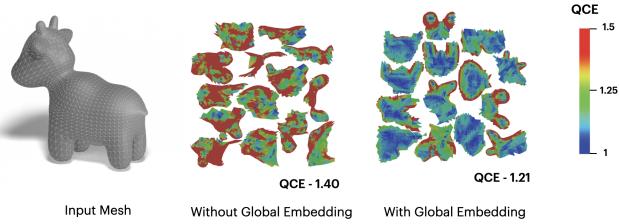


Figure 4: Use of Diffusion Net features provides global context to the parameterization MLP and removing it results in drastic increase of Quasi Conformal Error (QCE).

Effect of No of Patches on Parameterization: Figure 6 shows the trade-off between distortion (value of QCE and ASE) and the number of patches. With the increase in the number of patches, the distortion follows an up and down curve but eventually reduces significantly.

4.3 Effect of Loss Functions:

The cycle loss \mathcal{L}_{cycle} is crucial for self-supervised training of the parameterization module. However, it is not sufficient to get desired properties (isometry, conformality etc.). Figure 3 demonstrate the effect of additional loss functions, where the QCE value drops when \mathcal{L}_{iso} is introduced into the objective function, and it drops further when \mathcal{L}_{angle} & \mathcal{L}_{area} are also included.

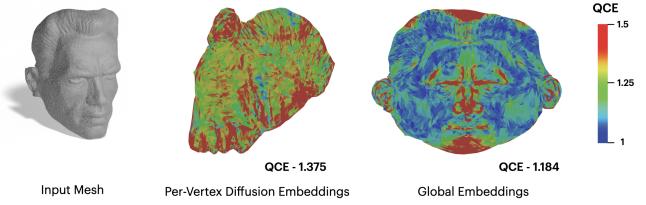


Figure 5: Comparison of global and local diffusion embedding inputs to forward MLP. Although diffusion is global process, feeding per-vertex features results in depletion of output quality.

5 LOSSES

The final objective function for surface parameterization is given as:

$$\mathcal{L}_{param} = \lambda_1 \mathcal{L}_{cycle} + \lambda_2 \mathcal{L}_{iso} + \lambda_3 \mathcal{L}_{angle} + \lambda_4 \mathcal{L}_{area} \quad (1)$$

The cycle consistency loss \mathcal{L}_{cycle} imposes bijectivity constraints in the UV space, while the isometric loss \mathcal{L}_{iso} imposes isometry. The isometric loss \mathcal{L}_{iso} is designed to impose isometric constraint in the UV space. Inspired by [Zigelman et al. 2002], the loss ensures that the geodesic distance between a pair of vertices in 3D space $G_d \in \mathbb{R}^{V \times V}$ is equal to the euclidean distance $E_d \in \mathbb{R}^{V \times V}$ in the UV space. The \mathcal{L}_{iso} is given as:

$$\mathcal{L}_{iso} = \|G_d, E_d\| \quad (2)$$

where $\|\cdot\|$ represents the L_2 norm. This loss is imposed only on geodesic distances less than a certain threshold σ . We choose $\sigma = 0.2$ for all our experiments.

We use \mathcal{L}_{angle} loss to reduce conformal error in the UV space. We take an L_2 norm between the angles $\theta_{i=1}^3$ per face belonging to F in the 3D space and faces f in UV space, given as:

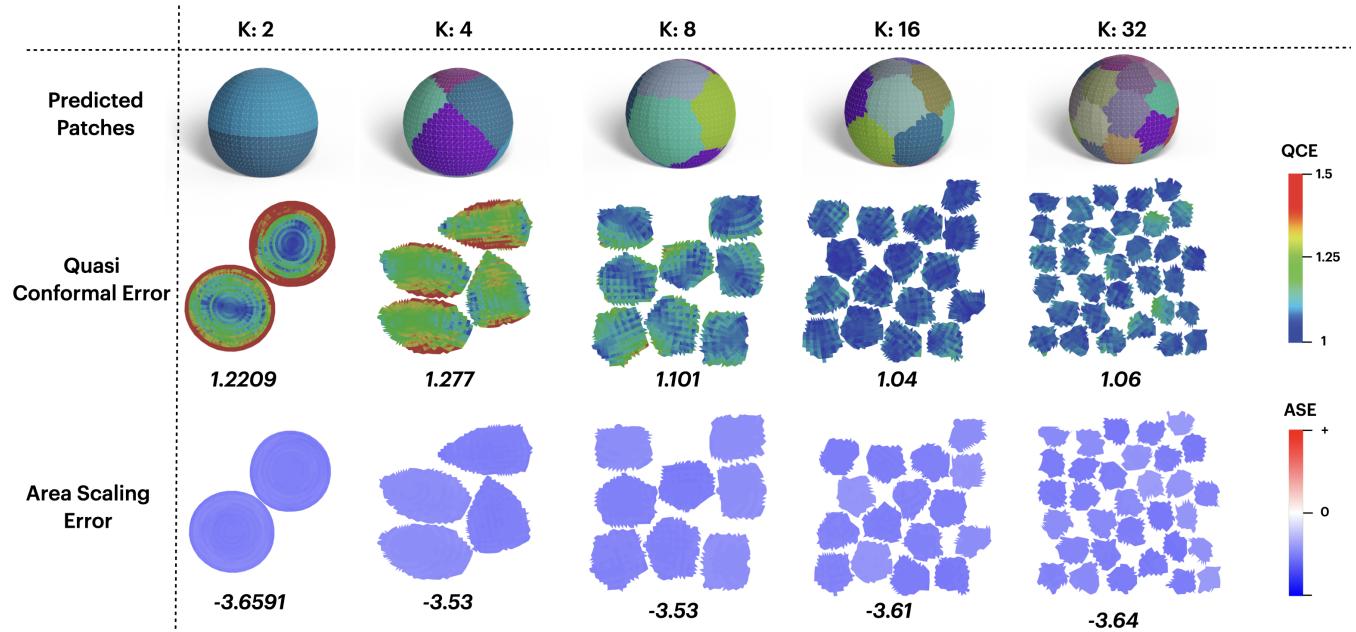


Figure 6: Patches are used to obtain multiple open surfaces from closed surfaces. As we increase the number of patches, the conformal and angular distortion gets reduced.

$$\mathcal{L}_{angle} = \frac{1}{|F|} \sum_{j=1}^{|F|} \frac{1}{3} \sum_{i=1}^3 \left(\cos(f_{\theta_i}^j) - \cos(F_{\theta_i}^j) \right)^2 \quad (3)$$

where $|F|$ is total the number of faces.

Similarly, \mathcal{L}_{area} loss is used to minimise the area distortion by taking an L_2 norm between the areas a_p, a_q of the faces f in the 3D space and faces F in UV space, respectively. The loss is given as follows:

$$\mathcal{L}_{area} = \frac{1}{|F|} \sum_{a_p, a_q \in f, F} \left(a_p - a_q \right)^2 \quad (4)$$

6 EVALUATION METRIC

We use Quasi Conformal Error (QCE) [Sander et al. 2001] and Area Scaling Error (ASE) [Sawhney and Crane 2017] for evaluation of the UV distortions. QCE measures the angular distortion based on the ratio of the singular values of each face mapping. The ideal QCE value is 1, and a higher value implies distortion. ASE measures the scale factor of the mapped faces. Negative ASE values imply shrinkage; positives imply increase, and zero implies no area distortion in mapping.

REFERENCES

- Noam Aigerman, Kunal Gupta, Vladimir G. Kim, Siddhartha Chaudhuri, Jun Saito, and Thibault Groueix. 2022. Neural Jacobian Fields: Learning Intrinsic Mappings of Arbitrary Meshes.
- Zhiqin Chen, Kangxue Yin, and Sanja Fidler. 2022. AUV-Net: Learning Aligned UV Maps for Texture Transfer and Synthesis.
- Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan C. Russell, and Mathieu Aubry. 2018. AtlasNet: A Papier-Mâché Approach to Learning 3D Surface Generation. (2018).

- jpcy. 2022. xatlas. <https://github.com/jpcy/xatlas>
- Bruno Lévy, Sylvain Petitjean, Nicolas Ray, and Jérôme Maillot. 2002. Least Squares Conformal Maps for Automatic Texture Atlas Generation. *ACM Trans. Graph.* (2002).
- Minchen Li, Danny M. Kaufman, Vladimir G. Kim, Justin Solomon, and Alla Sheffer. 2018. OptCuts: Joint Optimization of Surface Cuts and Parameterization. *ACM Transactions on Graphics* 37, 6 (2018).
- Pedro V. Sander, John Snyder, Steven J. Gortler, and Hugues Hoppe. 2001. Texture Mapping Progressive Meshes. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '01)*. Association for Computing Machinery, New York, NY, USA, 409–416. <https://doi.org/10.1145/383259.383307>
- Rohan Sawhney and Keenan Crane. 2017. Boundary First Flattening. *ACM Trans. Graph.* (2017).
- Nicholas Sharp, Souhaib Attaiki, Keenan Crane, and Maks Ovsjanikov. 2020. DiffusionNet: Discretization Agnostic Learning on Surfaces. (2020).
- He Wang, Kirill A. Sidorov, Peter Sandilands, and Taku Komura. 2013. Harmonic Parameterization by Electrostatics. *ACM Trans. Graph.* (2013).
- Francis Williams, Teseo Schneider, Claudio Silva, Denis Zorin, Joan Bruna, and Daniele Panozzo. 2018. Deep Geometric Prior for Surface Reconstruction. (2018).
- G. Zigelman, R. Kimmel, and N. Kiryati. 2002. Texture mapping using surface flattening via multidimensional scaling. *IEEE Transactions on Visualization and Computer Graphics* 8, 2 (2002), 198–207. <https://doi.org/10.1109/2945.998671>