Pacific Graphics (2019)
N. N, N. N, and N. N (Editors)

Supplemental materials

Connectivity-preserving Smooth Surface Filling with Sharp Features

Thibault Lescoat[1] Pooran Memari[2,3] Jean-Marc Thiery[1] Maks Ovsjanikov[2] Tamy Boubekeur[1]
[1] LTCI, Télécom Paristech, Institut Polytechnique de Paris [2] LIX, École Polytechnique, Institut Polytechnique de Paris [3] CNRS

1. Feature lines

Detection. We first detect feature points on $\mathcal{M}$: these are the boundary vertices where the surface forms a sharp edge. More precisely, for each successive boundary vertices (u,v,w), where boundary edges (u,v) and (v,w) are contained in triangles t_1 and t_2 with normals n_1 and n_2, we compute the following quantities (normalized vectors are noted $\widehat{x} = x/\|x\|$):

- $\theta_v = \operatorname{acos}(n_1 \cdot n_2)$, the angle between the triangles at v,
- d_v, the direction of the sharp edge:

$$d_v = \operatorname{sign}((n_1 \times (v-u)) \cdot s_v)\ \widehat{s_v}$$

$$\text{where } s_v = \begin{cases} n_1 \times n_2 & \text{if } |n_1 \cdot n_2| < 1 \\ n_1 \times (\widehat{v-u} + \widehat{w-v}) & \text{otherwise} \end{cases}$$

- $\tau_v = -\widehat{n_1 + n_2}$, the direction toward the interior of $\mathcal{M}$

We keep as feature points only boundary vertices v for which $\theta_v \geq \theta_{min}$, with θ_{min} a user parameter (see Figure 1).

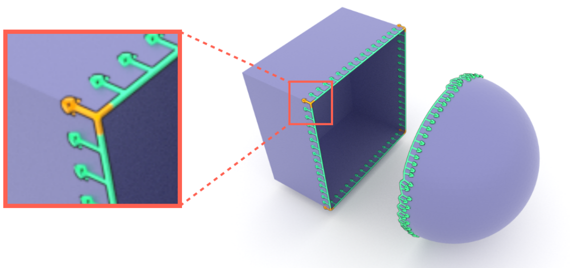

Figure 1: *For each boundary vertex, we compute a "sharp" direction with a fallback in case there is no sharp edge (arrows). Arrow colors reflect the angle between the boundary triangles.*

2. Results

Using the same regularization metric for both feature lines and the rest of the surface allows us to smoothly blend feature lines into the surface. This is especially desirable when a salient point is not paired with another but with the closest point on a different part of the input (see Figure 2 right).

When comparing with the method of Centin and Signoroni [CS18], we have the guarantee that the input connectivity is exactly preserved, along with the input geometry; furthermore, since we use the screened Poisson reconstruction only to unambiguously link the components of the input, we can use a lower depth, ensuring not only that the computation is faster, but also that it is more faithful (see Figure 4).

In parts of the input with numerous samples with opposite or orthogonal normals, the Poisson surface can exhibit tunnels (Figure 5 bottom), although this limitation is also relevant for the method of Centin and Signoroni [CS18] as it is also based on the Poisson reconstruction.

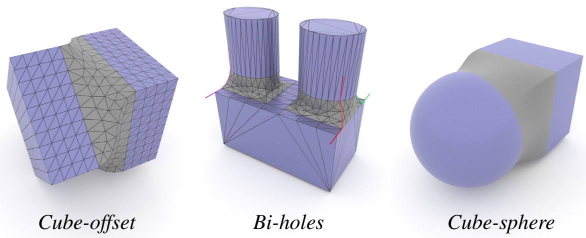

Figure 2: *Composition of simple shapes. We show the feature lines on bi-holes. Notice how they smoothly blend into the surface for cube-sphere.*

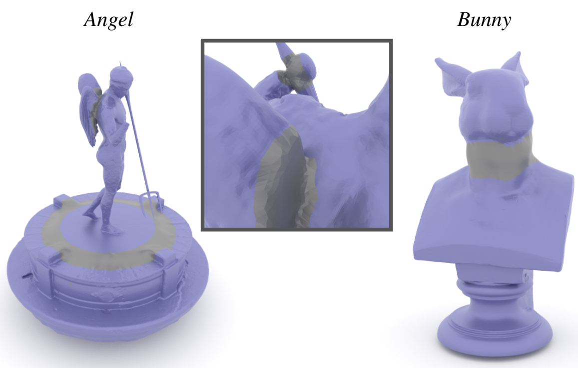

Figure 3: *Composition of complex shapes; input meshes are from EPFL, the Smithsonian 3D Digitization and Turbosquid.*

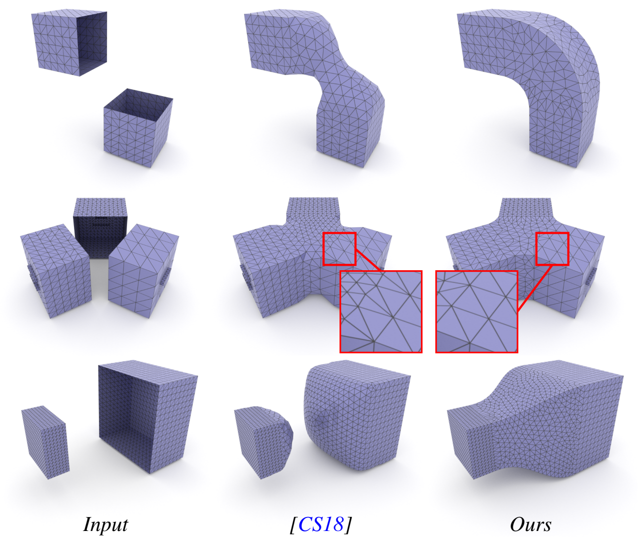

Figure 4: *Comparing our method to [CS18] for simple shapes. The input connectivity is modified for the latter; furthermore, the lack of resampling and the high depth used can result in disconnected components.*

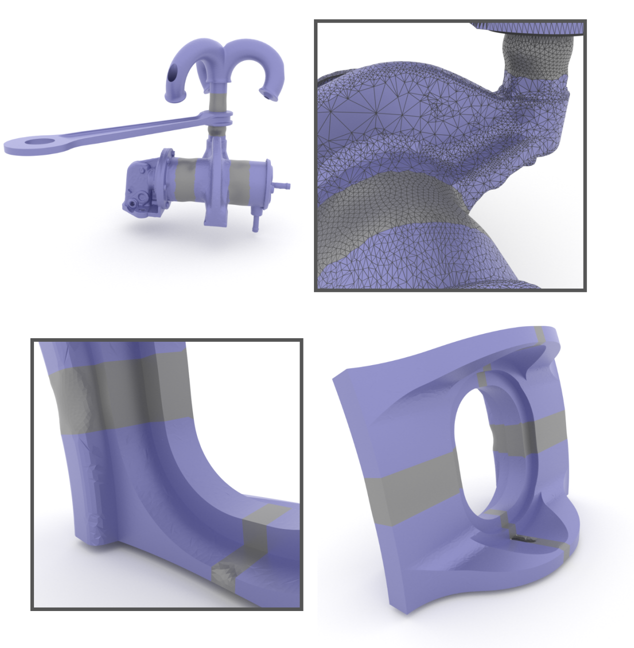

Figure 5: *Composition of complex shapes: pump on top, mech on bottom; the input meshes were obtained from Aim@Shape and Thingi10K [ZJ16]. The bump on pump is because of the very high curvature in the input mesh near the junction. The hole on mech comes from the Poisson implicit surface (and thus is also present with the Advancing Fronts method [CS18]).*

References

[CS18] Centin M., Signoroni A.: Advancing mesh completion for digital modeling and manufacturing. *Computer Aided Geometric Design 62* (2018), 73 – 90. 1, 2

[ZJ16] Zhou Q., Jacobson A.: Thingi10k: A dataset of 10,000 3d-printing models. *arXiv preprint arXiv:1605.04797* (2016). 2