# SBL Mesh Filter: Fast Separable Approximation of Bilateral Mesh Filtering

Guillaume Vialaneix
EDF R&D - Telecom ParisTech - CNRS

Tamy Boubekeur
Telecom ParisTech - CNRS

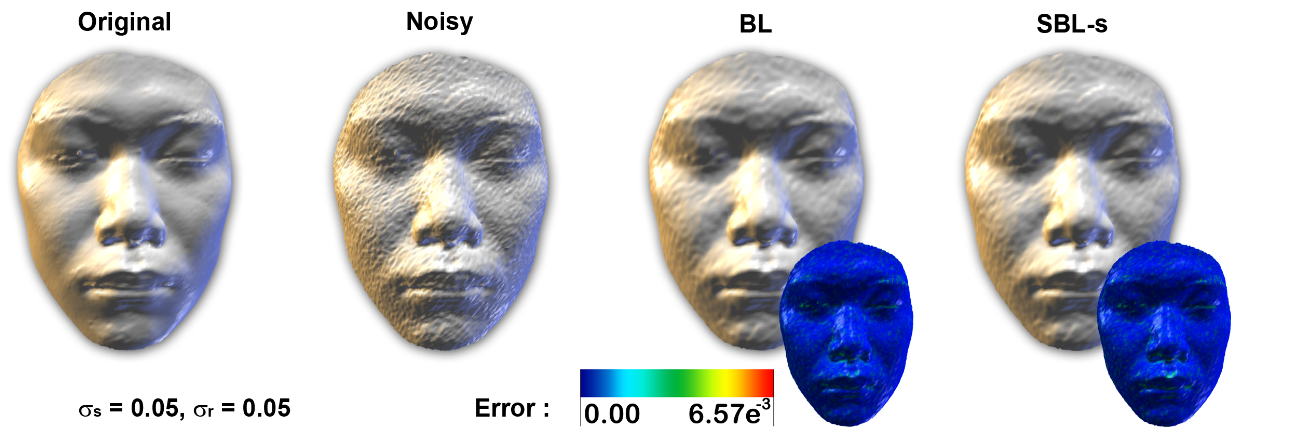


**Figure 1:** *From left to right : the original model, the model with noise addition, the BL denoising method (2sec), our SBL technique (1sec).*

**Introduction.** Bilateral mesh filtering (BL) is a simple and powerful feature-preserving filtering operator which allows to smooth or remove noise from surface meshes while preserving important features in a non-iterative way. However, to be effective, such a filter may require quite a large support size, inducing slow processing when applied on high resolution meshes. In this paper, we propose a separable approximation of BL (SBL) based on a local decomposition of the bi-dimensional filter into a product of two one-dimensional ones. The main problem here is to find meaningful directions at every point to orient the two one-dimensional filters. Our solution exploits the minimum and maximum curvature directions at each point and demonstrates a significant speed-up on meshes ranging from thousands to millions of elements, enabling feature-preserving filtering with large support size in a variety of practical scenarii (Fig. 1).

**Separable Bilateral Filtering.** Considering a noisy object, a simple low-pass filter helps reducing the noise, but it considers only the distances in space of neighboring samples to weight their contributions in a (local) combination. Consequently, noise is filtered out, but feature lines are proportionally blurred. The idea of bilateral filtering is to introduce a second weighting term based on the difference in range between object samples to weight their relative contribution. We focus here on Jones and Durand formulation [Jones and Durand 2003] of the BL. They define a range space for meshes by the mean of predictions computed as the projection of the vertex onto the tangent plane of its neighbors : considering a mesh $\mathcal{M}$, and one of its vertices $\mathbf{p}_i$, then for each of the faces $\mathcal{F}_j$ in its neighborhood $\mathcal{N}_i$, we can compute the projection $\mathcal{P}_{\mathbf{c}_j}(\mathbf{p}_i)$ of $\mathbf{p}_i$ on the plane defined by the center $\mathbf{c}_j$ and the *smoothed* normal $\mathbf{n}_j$ of $\mathcal{F}_j$. Using a spatial kernel $G_{\sigma_s}$ and a range kernel $G_{\sigma_r}$, a vertex $\mathbf{p}_i$ is filtered according to the neighboring faces set $\mathcal{N}_i$ by

$$BL_{\mathcal{N}_i}(\mathbf{p}_i) = \sum_{\mathcal{F}_j \in \mathcal{N}_i} G_{\sigma_s}(d_{ij}) G_{\sigma_r}(h_{ij})\, \mathbf{c}_j \tag{1}$$

with $d_{ij} = ||\mathbf{p}_i - \mathbf{c}_j||$ and $h_{ij} = ||\mathbf{p}_i - \mathcal{P}_{\mathbf{c}_j}(\mathbf{p}_i)||$. The key idea of our approximation model is to speed-up the BL filter by reducing the size of $\mathcal{N}_i$ while still covering the same support size. Our approximation works in two passes: in the first pass we collect a set of neighboring faces restricted to one tangent direction on the surface and then filter the vertex using this reduced set only. This first pass is applied to all vertices. In the second pass, we filter the output of the first pass using an orthogonal tangent direction. This approach is inspired by the classical separable Gaussian filter for images. Note however that while the exact solution is reproduced in the case of Gaussian filtering, such a decomposition leads only to an approximate feature-preserving filter [Pham and van Vliet 2005]. Thus, we compute filtered minimum and maximum curvature directions $\{\mathbf{u}_i, \mathbf{v}_i\}$ at every vertex $\mathbf{p}_i$ of $\mathcal{M}$ to define locally the filtering direction for our two passes. We define two orthogonal planes $\Pi^i_{\mathbf{u}} = \{\mathbf{n}_i, \mathbf{u}_i\}$ and $\Pi^i_{\mathbf{v}} = \{\mathbf{n}_i, \mathbf{v}_i\}$, one for each curvature direction and both intersecting along the vertex normal (Fig. 2). These planes intersect the neighboring faces and offer a straightforward predicate to collect the restricted set of faces for each pass, defined as the intersection of the full ball-neighborhood $\mathcal{N}_i$ with the planes $\mathcal{N}^{\mathbf{u}}_i = \left\{\mathcal{F}_j \in \mathcal{N}_i \quad | \quad \mathcal{F}_j \cap \Pi^i_{\mathbf{u}} \neq \varnothing\right\}$. Finally, our SBL approximation consist in the combination of a two restricted BL filtering:

$$SBL(\mathbf{p}_i) = BL_{\mathcal{N}^{\mathbf{u}}_i}\left(BL_{\mathcal{N}^{\mathbf{v}}_i}(\mathbf{p}_i)\right) \approx BL_{\mathcal{N}_i}(\mathbf{p}_i) \tag{2}$$

**Results.** Our SBL approximation model is twice or more faster with typical setting, while preserving a RMS error below 0.1% of the mesh size when compared to the exact filter. We show that our choice based on curvature directions is superior to other ones such as random or regular local frames, providing results which are visually indistringuishable from the exact BL filter.

## References

JONES, T. R., AND DURAND, F. 2003. Non-iterative, feature-preserving mesh smoothing. *ACM Transactions on Graphics 22*, 943–949.

PHAM, T., AND VAN VLIET, L. 2005. Separable bilateral filtering for fast video preprocessing. In *IEEE ICME 2005*.

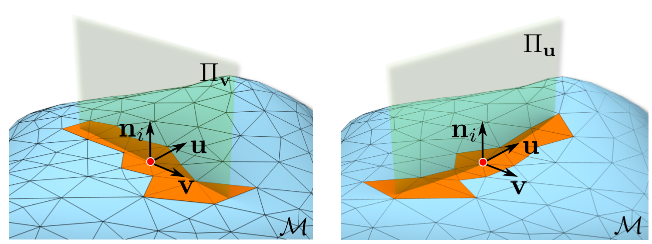


**Figure 2:** *Two local planes $\Pi^i_{\mathbf{u}}$ and $\Pi^i_{\mathbf{v}}$ (green), allow to query for two restricted sets of neighborhing faces (orange), one for each pass, which are aligned with curvature directions.*