Guided Mesh Normal Filtering

Wangyu Zhang  USTC
Bailin Deng    EPFL, University of Hull
Juyong Zhang   USTC
Sofien Bouaziz EPFL
Ligang Liu     USTC
Filtering is necessary
Related Work

- Bilateral mesh denoising [Fleishman et al. 2003]
- Non-iterative, feature-preserving mesh smoothing [Jones et al. 2003]
- Bilateral Normal Filtering for Mesh Denoising [Zheng et al. 2011]
- Mesh Denoising via L0 Minimization [He & Schaefer 2013]
Bilateral filter

\[ q_i = \sum_{j \in N(i)} W_{ij}(p) p_j \]

- **Bilateral filter**
- input \( p \)
- spatial \( G_s(x_i-x_j) \)
- range \( G_r(p_i-p_j) \)
- bilateral \( W = G_s G_r \)
- output \( q \)
Joint bilateral filter

\[ q_i = \sum_{j \in N(i)} W_{ij}(I) p_j \]

bilateral filter: \( I = p \)

- Joint bilateral filter [Petschnigg et al. 2004]

- \( W = G_s G_r \)

- \( G_s(x_i - x_j) \) spatial

- \( G_r(I_i - I_j) \) range

- input \( p \)

- guide \( I \)

- output \( q \)

E.g. \( p \): noisy / chrominance channel

\( I \): flash / luminance channel
The role of guidance

• The success of joint bilateral filtering is heavily dependent on the guidance signal.

• The guidance signal should provide a robust estimation about the features of the output signal.
The importance of guidance

Original

Noisy

With Ground Truth Normals

With Noisy Normals
Guidance geometry

• Contrary to the case of images, such guidance geometry is not easily available from measure devices.

• It often has to be constructed computationally.
Example

Noisy

With Ground Truth Normals

With Noisy Normals

Ours
Denoising pipeline
Guidance normal computation
Normal consistency

For each candidate patch $\mathcal{P} \in \mathcal{C}(f_i)$

we measure the consistency of its normals using

$$\mathcal{H}(\mathcal{P}) = \Phi(\mathcal{P}) \cdot \mathcal{R}(\mathcal{P})$$
Maximum normal difference

\[ \Phi(\mathcal{P}) = \max_{f_j, f_k \in \mathcal{P}} \| n_j - n_k \| \]
Edge saliency measurement

\[ R(\mathcal{P}) = \frac{\max_{e_j \in E_\mathcal{P}} \varphi(e_j)}{\varepsilon + \sum_{e_j \in E_\mathcal{P}} \varphi(e_j)} \]

\[ \varphi(e_j) = \| n_{j_1} - n_{j_2} \| \]
Patch selection
Normal filtering

• Our normal filtering computes a new unit normal for each face via joint bilateral filter:

\[
\overline{n}_i = \frac{1}{W_i} \sum_{f_j \in \mathcal{N}_i} A_j \ K_s(c_i, c_j) \ K_r(g_i, g_j) \ n_j
\]
Updating vertices

- Based on the filtering face normals, the vertex positions are updated by minimizing the $\ell_2$ error of the compatibility conditions:

$$\overline{n}_i \cdot (\overline{v}_{i_k} - \overline{c}_i) = 0 \ (k = 1, 2, 3)$$
Recap: pipeline
Denoising process

Denoising Process

Input

Error Metric $\ell_2$

#Iter
Results: vs bilateral filter

Input

[ZFAT11]

Ours
Results: comparisons

Noisy

[FDCO03] [JDD03] [SRML07] [ZAFT11] (local)

[ZAFT11] (global) [HS13] [WYP*15] Ours
## Time statistics

<table>
<thead>
<tr>
<th>Model</th>
<th>#Vertices</th>
<th>#Faces</th>
<th>Time(s)/Iter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fandisk</td>
<td>6475</td>
<td>12946</td>
<td>0.076</td>
</tr>
<tr>
<td>Block</td>
<td>8771</td>
<td>17550</td>
<td>0.104</td>
</tr>
<tr>
<td>Bunny</td>
<td>34834</td>
<td>69451</td>
<td>0.698</td>
</tr>
<tr>
<td>Iron</td>
<td>85574</td>
<td>168285</td>
<td>1.571</td>
</tr>
</tbody>
</table>
Conclusion

- A joint bilateral filter for mesh processing
  - A novel method to construct the guidance signal
  - Effective and efficient, simple to implement
  - Much better denoising results than state of the art
The source code is available: https://github.com/bldeng/GuidedDenoising

Guided Denoising

This code implements the denoising algorithm in the following paper:


It also implements the denoising methods from the following papers for comparison:
Thank you!