Mesh Smoothing based on Anisotropic Mean Curvature Flow

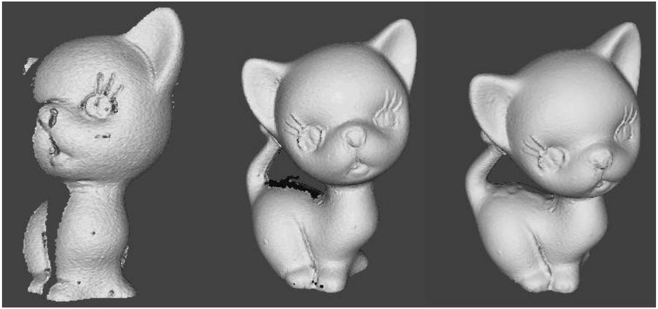
Phan-Anh Nguyen[1]

[1]Chair of Computer Graphics & Multimedia
RWTH-Aachen University

Geometry Processing Lab, 2012

Outline

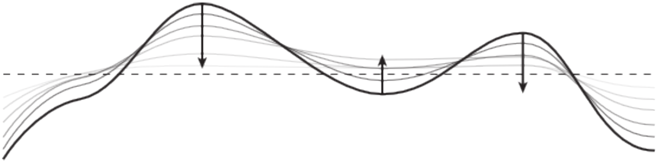
Geometry pipeline

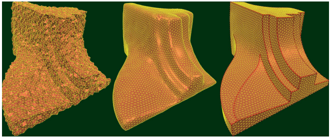


- Many range image patches are acquired by a 3D scanner
- Patches are aligned and merged
- Holes are filled and the mesh is **smoothed**

Remove noise from 3D scaners

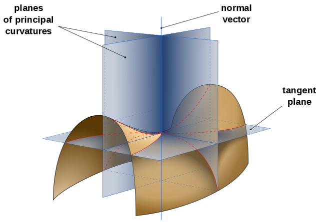
- Idea: Diffuse high curvature region by averaging over its neighborhood.

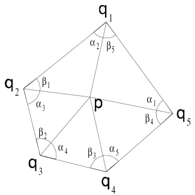




- Isotropic e.g. Laplacian: sharp features not preserved
- Anisotropic mean curvature: preserves sharp features

Mean Curvature





- Curvature: measure deviation from flat
- Mean Curvature: average between min and max curvatures
- Smooth: reduce curvature (equivalent to area gradient)

$$\bigtriangledown_p \, area \ M = \vec{H}(p) = 1/2 \sum_{q_i \in link \ p} (cot\alpha_i + cot\beta_i)(p - q_i)$$

Mean Curvature

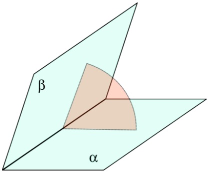


Figure : Dihedral angle θ_e

- Let $\vec{H}(e) = H_e\vec{N}_e$ be an edge mean curvature vector, where $\vec{N}_e$ is edge normal and $H_e = 2|e|cos(\theta_e/2)$ then:

$$\vec{H}(p) = 1/2 \sum_{e=(p,q),q\in link\ p} \vec{H}(e)$$

Anisotropic Mean Curvature

- Weight less for feature vertices to avoid smoothing sharp features

$$\vec{H}_A(p) = 1/2 \sum_{e=pq, q \in link\ p} w(H_e) H_e \vec{N}(e)$$

$$w_{\lambda,r}(a) = \begin{cases} 1 & \text{for } |a| \leq \lambda \\ \dfrac{\lambda^2}{r(\lambda - |a|)^2 + \lambda^2} & \text{for } |a| > \lambda \end{cases}$$

Anisotropic mean curvature flow

- Explicit iteration step of the anisotropic mean curvature flow

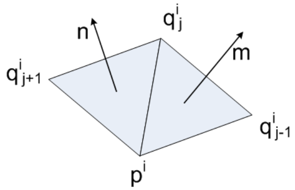
$$p^{j+1} = p^j - \frac{3s}{area(star\ p^j)} \vec{H}_A(p^j)$$

- s controls the speed of an integration step
- Matrix form:

$$\mathcal{P}^{j+1} = \mathcal{P}^j - sM^{-1}\vec{H}_A(\mathcal{P}^j)$$

$$M_{pq} = \begin{cases} \frac{1}{6} area(starp), & \text{if } p = q \\ \frac{1}{12} area(stare), & \text{if there is an edge } e = (p, q) \\ 0, & \text{otherwise} \end{cases}$$

Anisotropic mean curvature - Matrix form



$$
\begin{aligned}
\vec{H}_A(p^i) &= 1/2 \sum_{q^i_j \in N(p^i)} w(H_{e_j}) H_{e_j} \frac{1}{\|\mathbf{n}+\mathbf{m}\|}(\mathbf{n}+\mathbf{m}) \\
&= \sum_{q^i_j \in N(p^i)} w_j \left[(q^i_j - p^i) \times (q^i_{j+1} - p^i) + (q^i_{j-1} - p^i) \times (q^i_j - p^i) \right] \\
&= \sum_{q^i_j \in N(p^i)} w_j \left[(q^i_j - p^i) \times (q^i_{j+1} - q^i_{j-1}) \right] \\
&= \sum_{q^i_j \in N(p^i)} w_j \left[A_\times \cdot q^i_{j+1} - A_\times \cdot q^i_{j-1} \right]
\end{aligned}
$$

Drawback of Anisotropic mean curvature

- slow down the smoothing process in regions with high curvature.
- cause deformations of the surface

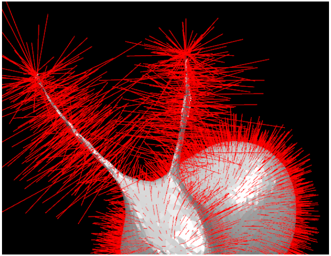


Figure : anisotropic mean curvature after 100 iterations

Prescribed Mean Curvature

- AMC: $\vec{H_A}(p) \to 0$, APMC: $\vec{H_A}(p) \to H\vec{V_A}(p)$, where H is some constant mean curvature
- compute mean curvature (H), smooth this scalar field
- evolve the surface towards a surface having this smoothed mean curvature

$$p^{j+1} = p^j - \frac{3s}{area(star\ p^j)}(\vec{H_A}(p^j) - f(p^j)\vec{V_A}(p^j))$$

- f is a function, that prescribes the anisotropic mean curvature
- $\vec{V_A}$ is an anisotropic volume gradient.

Results

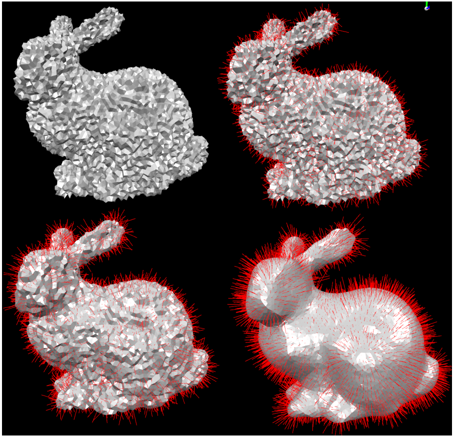


Figure : bunny after 0, 5, 10, 30 iterations

Results

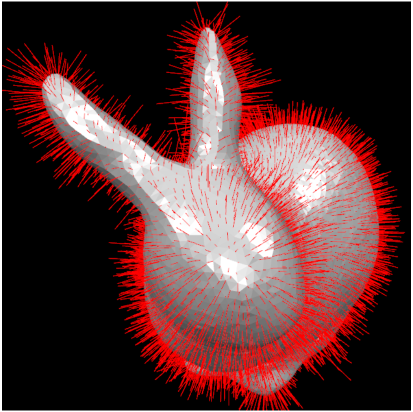


Figure : after 100 iterations

Results

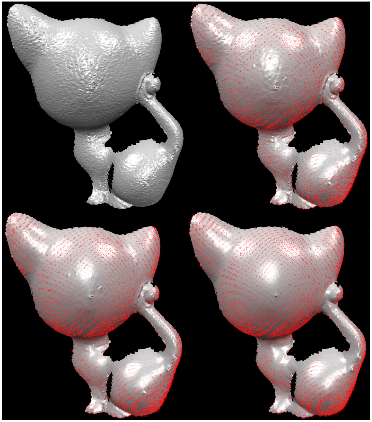


Figure : after 0, 5, 10, 100 iterations

Results

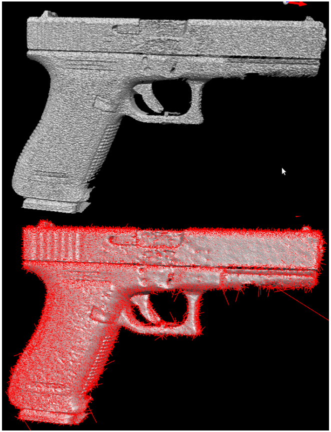


Figure : after 0, 20 iterations

Summary

- Anisotropic mean curvature: reduce smoothing in high curvature regions
  - Preserve linear features but leads to deformation of curves
- Anisotropic prescribed mean curvature: evolve the surface to the precomputed, smoothed mean curvature

- Future work
  - Implement implicit integration methods
  - Color coding
  - Extend to process boundaries