Point Cloud Denoising

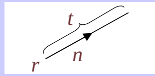
Luiz Velho
Boris Mederos Madrazo
Luiz Henrique de Figueiredo

IMPA – Instituto de Matemática Pura e Aplicada

Problem Statement

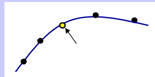
Surface Reconstruction and Denoising

$$Q(r) = r + t_r \cdot n_r$$



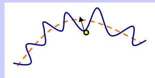
- Projection Operator

$$\{x \mid Q(x) = x\}$$



- Smoothing Operator

$$x' \leftarrow Q(x)$$



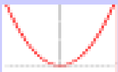
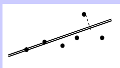
Surface Estimation

Minimize Error Norm: $E(x)$

$E(x)$ S

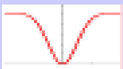
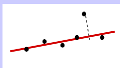
- Non-Robust Statistics
  - Least Squares

$$g(x) = x^2$$

- Robust Statistics
  - W-Estimator

$$\rho_\theta(x) = \sigma^2(1 - e^{-x^2/2\sigma^2})$$

  - M-Estimator

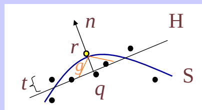
$$\theta^*(x) = \arg\min_\theta \sum \rho(x,\theta)$$

Related Work I

- *Moving Least Squares* (Levin *et al.*)

$$S = \{x \in \Re^3 \mid \psi(x) = x\}$$

$$\psi(x) = r + [t + \gamma(0,0)]n$$



- Reference Plane: $H_r = (n,t)$

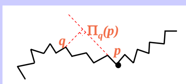
$$(n,t) = \arg\min_{n,t} \sum_{p\in \mathbf{N}(r)} (n\cdot(p-r-tn))^2 W_h(\| p-r-tn \|)$$

- Height Polynomial: $\gamma(u) = \sum_{j=0}^{n} a_j u^j$

$$\min_{a_0,\dots,a_n} \sum_{p\in S} (\gamma(u_p) - h_p)^2 W_h(\| p - r - tn \|)$$

Related Work II

- *Bilateral Filtering*



minimize energy fitting $E(p)$

$$E(p) = \int_{q\in S} W_c(\| q-p \|)\rho(\| \Pi_q(p) - p \|)dq$$

- Jones *et al.*

$$p' = \frac{1}{k(p)} \sum_{q\in\Omega} \Pi_q(p) W_c(\| c_q - p \|) W_s(\| \Pi_q(p) - p \|) a_q$$

- Fleishman *et al.*

$$p' = p + \left[\frac{1}{l(p)} \sum_{q\in N(p)} W_c(\| p-q \|) W_s(<n, p-q>) \right] \cdot n$$

➢ *Obs: normal estimation*

Proposed Method

Given a Point Cloud

$$C = \{p \in \Re^3 \mid p = q + \varepsilon, \quad q \in S, \quad \varepsilon \sim \mathbf{N}(0,\sigma)\}$$

Compute Operator

$$r^* = Q(r) = r + t_r n_r \qquad \text{with} \quad \| n \|^2 = 1$$

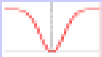
by Solving Equation (1)

$$\{n^*, t^*\} = \arg\min_{\{n,t\}} \sum_{q\in N_p} g(n^T(q-p) - t) w(\| q - p \|)$$

Adaptive, Feature Preserving Reconstruction

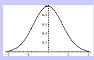
- Robust Error Norm

$$g(x) = \sigma_g \left(1 - \exp\left(\frac{-x^2}{2\sigma_g}\right) \right)$$



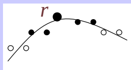
- Smoothing Weight

$$w(x) = \exp\left(\frac{-x^2}{2\sigma_w}\right)$$



- Local Neighborhood (*progressive propagation*)

$$N_\sigma(p) = \{ s_i ; \cup\, s \in K_\sigma(s_i) \wedge \| s - p \| < \sigma_N \}$$



Main Algorithm

- Iterative Approach
  - Alternate Minimization with Respect to ***n*** and ***t***

t_0 = 0
k = 0
repeat {
  (i) n_{k+1} = minimum of Eq.(1) with respect to n with fixed t = t_k
  (ii) t_{k+1} = minimum of Eq.(1) with respect to t with fixed n = n_{k+1}
  k = k +1;
} **until** (close to the minimum)
$p' = p + t_k n_k$

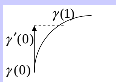
Stage (i)

- Constrained Optimization Problem

$$n^* = \arg\min_n \sum_{q \in N_p} g(n^T(q - p) - t)\, w(\| q - p \|)$$

subject to $\| n \|^2 = 1$

- Solve with Newton Method in ***S***^{2}



  - Geodesic Path

$$\gamma(0) = n_i, \quad \gamma(1) = n_{i+1}, \quad \gamma'(0) = H_i$$

Newton Method on Riemannian Manifolds

- (Smith, 1994)

Select initial point $n_1 \in M$ such that $(\nabla^2 f)_{n_1}$ is non-degenerate
i = 1
repeat {
  Compute $H_i = -(\nabla^2 f)^{-1}_{n_i} (\nabla f)_{n_i}$
  Find $n_{i+1} = \exp_{n_i} H_i$
  i = i + 1
} **until** (close to the minimum)

with $f : M \to \Re$

✓ *Quadratic Convergence*

➢ *Need a Good Starting Point*

Stage (ii)

- Unconstrained Optimization Problem
  - First Order Condition

$$\sum_{q \in N_p} \Psi_w(\| q - p \|)\Psi_g(h_q)\ (n^T(q - p) - t) = 0$$

with $\Psi_g(x) = \frac{1}{x\sigma_g} g'(x)$, $\Psi_w(x) = w(x)$, and $h_q = n^T(q - p) - tn$

  - Solving for *t* (recurrence relation)

$$t_{i+1} = \frac{1}{k_{t_i}} \sum_{q \in N_p} \Psi_w(\| q - p \|)\Psi_g(n^T(q - p - t_i n))\, n^T(q - p)$$

✓*Convergence*

Convergence Analysis

Lemma: *The sequence* t_i *is strictly monotone decreasing* ($t_i > t_{i+1}$) *and converges to the minimum of the function:*

$$F_n(t) = \sum_{q \in N_p} g(h_q - t)\, w(\| q - p \|)$$

Proof: *This sequence* t_i *is a particular case of an M-estimator*

➢ **Note**: Fleishman *et al*. method corresponds to the first iteration of the above algorithm, starting at $t_0 = 0$.

$$t_1 = \frac{\sum_{q \in N_p} W_g(h_q) W(\| q - p \|)\, h_q}{\sum_{q \in N_p} W_g(h_q) W(\| q - p \|)}$$

Initialization

➢ *Need Good Estimate of the Normal for Stage (i)*

- Robust Method to Compute Normals of ***C***
  - Make $t = 0$ in Eq. (1) and Compose the Lagrange Equation

$$L(n,\lambda) = \sum_{q\in N_p} w(\| q-p \|)\, g(n^T(q-p)) + \lambda(\| n \|^2 - 1)$$

  - Compute the derivative w.r.t. n, and set it to zero

$$L_n(n,\lambda) = \sum_{q\in N_p} \Psi_w(\| q-p \|)\Psi_g(h_q)(q-p)(q-p)^T n - \lambda n = 0$$

Matrix Solution

- Rewrite previous Equation as:

$$M(n)\, n = \lambda\, n$$

where

$$M(n) = \sum_{q\in N_p} \Psi_w(\| q-p \|)\Psi_g(n^T(q-p))(q-p)(q-p)^T$$

- Iterative Scheme

$$M(n_k)\, n_{k+1} = \lambda_{k+1}\, n_{k+1}$$

λ_{k+1}, n_{k+1} smallest eigenvalue, eigenvector of $M(n)$

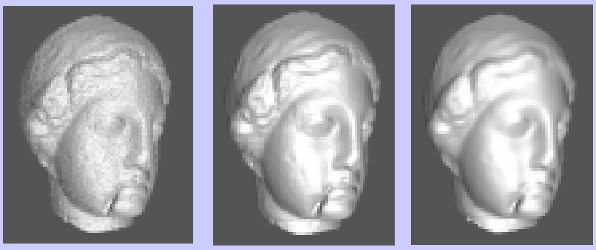
✓ *Good results in few iterations*

Experiments

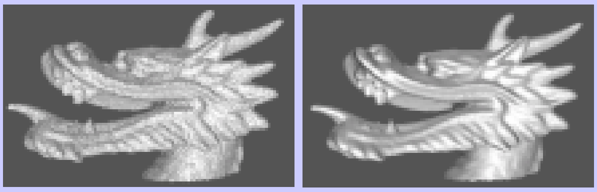
- Tested on Several Datasets
  - Polygonal models, discarding connectivity (*for comparisons*)
- Corrupted Models
  - Gaussian noise (with variance 0.02) in the normal direction
- Parameters
  - $\sigma_g = 1.5h,\ 3h$
  - $\sigma_w = 2h,\ 4h$
  - h: mean spacing between sample points

Bunny

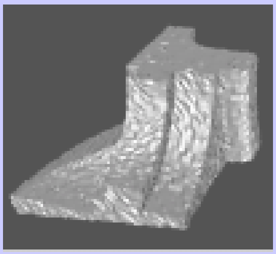
Igea

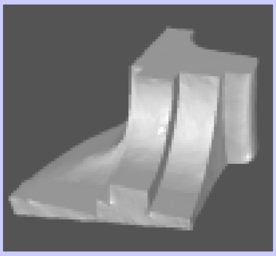


Dragon



Fandisk





Performance

- Convergence to 10^{-10}
  - Stage (i) : average of *6* iterations
  - Stage (ii): average of *9* iterations
- Timings for Bunny model
  (Pentium IV, 1.6 GHz, 512 Mb RAM)
  - $\sigma_w = 2h,$: *9.91* sec
  - $\sigma_w = 4h$: *19.67* sec

Comparison

Proposed Method		**Bilateral Filter**
Iterative	⟷	Non Iterative
M-estimator and *W-estimator*	⟷	*W-estimator*

Proposed Method		**Moving Least Squares**
Robust Statistics	⟷	Non-Robust Statistics
Two parameters	⟷	One parameter
Local Neighborhood	⟷	Global Neighborhood
One step	⟷	Two steps

Conclusions

- Modified MLS to be Feature Preserving
- Extended Bilateral Filter to Point Sets
- Robust Surface (and Normal) Estimation
- Effective Numerical Optimization Methods