# 3D Scale-space for Point Sets

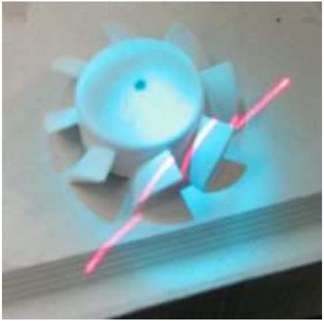
Julie Digne
julie.digne@cmla.ens-cachan.fr
Work under supervision of Jean-Michel Morel

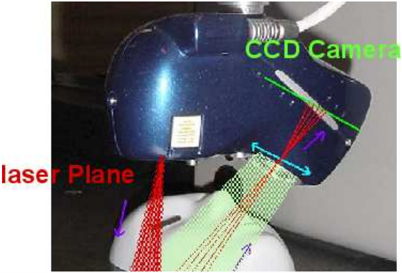
**CMLA - ENS Cachan**

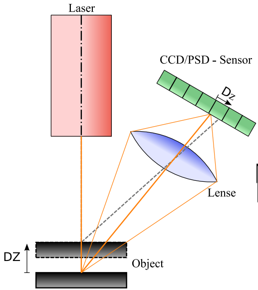
July, 8 2009
Recent Advances on Topological and Geometric Data Analysis Workshop

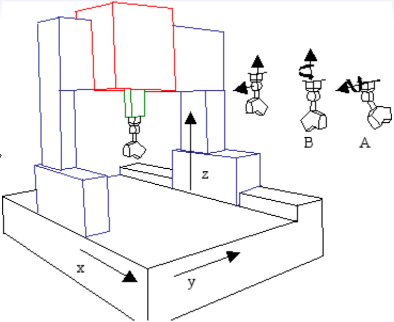
# Introduction

- Object acquisition by a high precision laser scanner
- Goal: accurate analysis of the object's geometry
- Application: automatic objects scanning by closed loop data analysis








**Triangulation laser scanner:** triangle formed by the camera optic center, the laser emitter and the impact point The result given by the scanner is a list of unoriented $3D$ points

- The initial dataset is a set of positions with no connectivity information

- The initial dataset is a set of positions with no connectivity information
- Additional constraint: holes should be detected but not filled, information should be given on the initial raw dataset

- The initial dataset is a set of positions with no connectivity information
- Additional constraint: holes should be detected but not filled, information should be given on the initial raw dataset
- Level sets methods ([HDD+92],[Kaz05],[KBH06]...) cannot be used

- The initial dataset is a set of positions with no connectivity information
- Additional constraint: holes should be detected but not filled, information should be given on the initial raw dataset
- Level sets methods ([HDD+92],[Kaz05],[KBH06]...) cannot be used
- **It leads us to a scale space approach**

# Outline

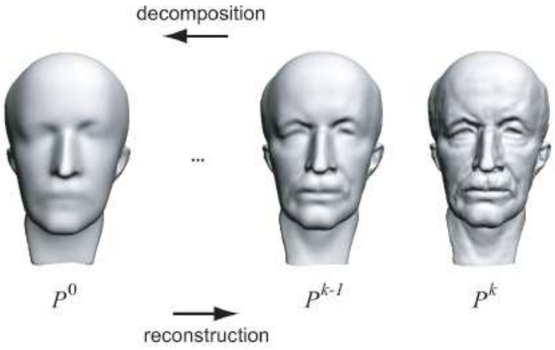
1. **Previous Work**

2. 3D Scale Space
   - Analogy in 2D image processing
   - 3D equivalent evolution equation

3. Application: Mesh reconstruction Problem

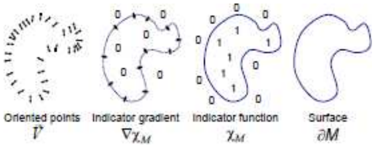
# Multiscale Methods

- Most multiscale approaches are defined on meshes
- In [PKG06], a curvature motion is defined and a back projecting operator is associated

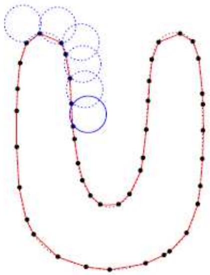


# Triangulation Methods

Triangulation of a set of points by level sets method ([HDD+92],[KBH06]) (approximating methods):
A signed distance function to the surface is defined and its 0 level set is sampled by the marching cube algorithm for example



Triangulation by Voronoi/Delaunay methods (interpolating methods)
Ball Pivoting Algorithm ([BMR+99]): A triangle between three points is created if the $r$ radius sphere going through those three points does not contain any other point.



# Outline

1. Previous Work
2. 3D Scale Space
   - Analogy in 2D image processing
   - 3D equivalent evolution equation
3. Application: Mesh reconstruction Problem

## Analogy in 2D image processing

Evolve an image by a PDE (e.g. heat equation), perform treatment at a low resolution and propagate the results to the original image.

3D equivalent evolution equation

Let $p$ be a point of a data set $\mathcal{M}$ and denote by $B_r(p)$ the set of all points $q$ in $\mathcal{M}$ such that $\|p - q\| < r$.

**Theorem**

*In the local intrinsic coordinate system of a continuous and smooth two-dimensional manifold $\mathcal{M}$, for $p \in \mathcal{M}$ the projection $p'$ of $p$ on the local regression plane has coordinates $x_{p'} = 0$, $y_{p'} = 0$ and $z_{p'} = \frac{Hr^2}{4} + o(r^2)$, where $H = \frac{k_1+k_2}{2}$ is the surface mean curvature at $p$ and $k_1$, $k_2$ the surface principal curvatures at $p$.*

## Consequence

- $H \approx \frac{4\langle p-p', \vec{n}(p)\rangle}{r^2}$
- It is a very stable estimate since it relies on order 1 approximation
- This yields our scale space operator:

**Theorem**

*Let $T_r$ be the operator defined on the surface $\mathcal{M}$ transforming each point $p$ into its projection on the local regression plane. Then*

$$T_r(p) - p = -\frac{Hr^2}{4}\vec{n}(p) + o(r^2). \tag{1}$$

*Thus, this operator is tangent to the mean curvature motion.*

# Estimated Curvature on a perfect torus

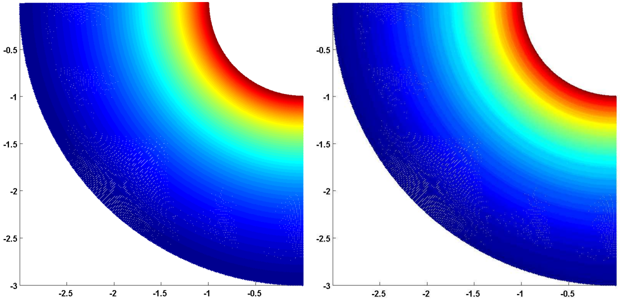
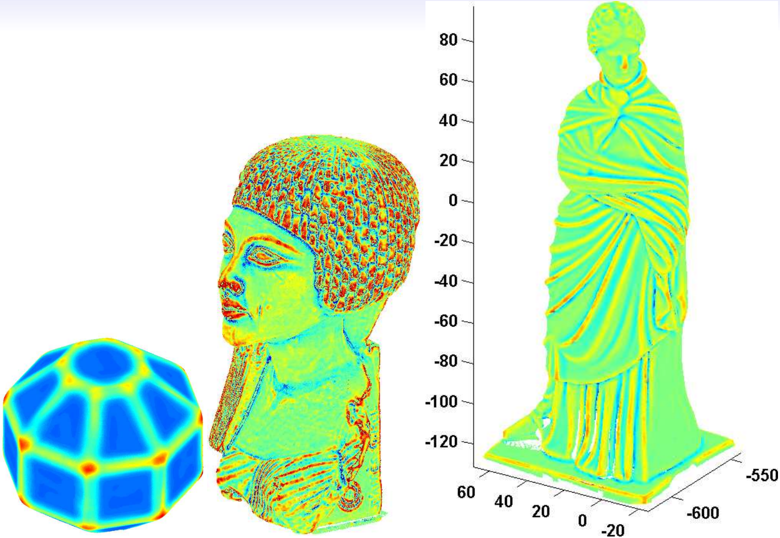



Figure: left: theoretical curvature, right: scale-space estimated curvature

# Estimated Curvature on various point sets



# Visualization of scale space effects

- A good way of seeing the evolution of a surface defined by a point cloud is to watch the evolution of level sets and especially positive and negative curvature level sets
- At first the level sets capture texture variations but when the surface is smoothed, is captures the object's shape

# Visualization of scale space effects

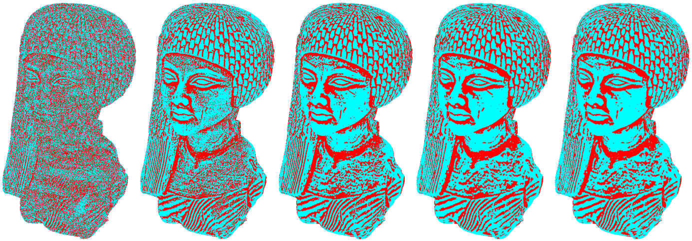
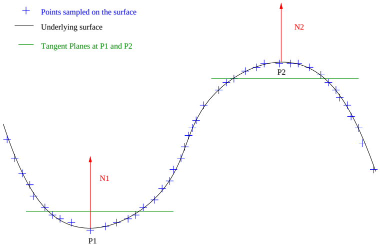


Figure: figs 1-4: 4 iterations of scale space are applied, fig. 5: by reverse scale space points are moved back to their original positions

# Outline

# Preprocessing: Normal estimation

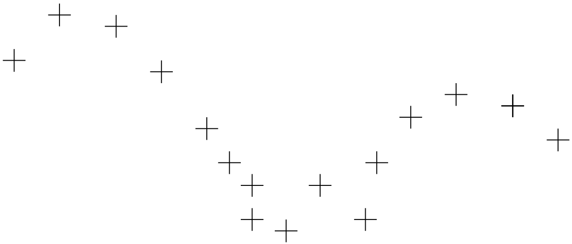
- For each point normal direction is known by neighborhood PCA
- Choose a point in a flat area, pick one of the two possible orientations
- Propagate orientation in neighborhoods



# Algorithm

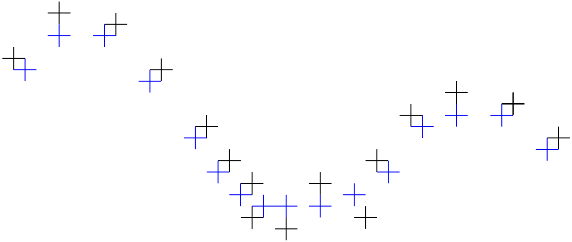
1. Iterate Projection Filter and keep a track at each step of the point displacement
2. Mesh the resulting samples. The obtained mesh is singularity free;
3. Project the mesh back to the original points.
4. The result is an interpolating mesh which preserves textures and details.

# synthetic 1D example



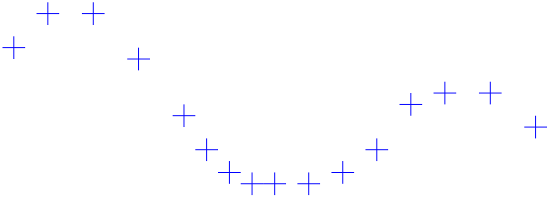
Initial points

# synthetic 1D example



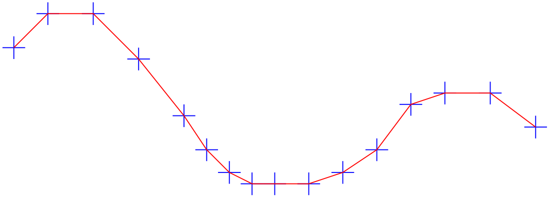
Initial points and their projections

# synthetic 1D example



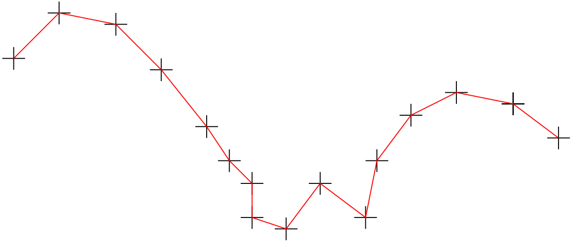
Projected points

# synthetic 1D example



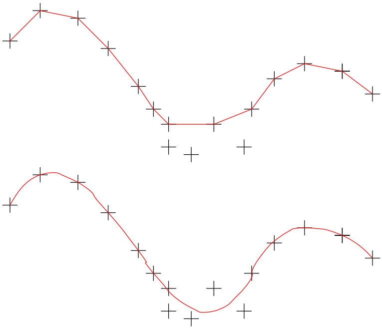
Resulting mesh of the projected points

# synthetic 1D example



Back projected mesh of the initial points

# synthetic 1D example



Top: same initial points with direct BPA triangulation

Bottom: same initial points with Poisson Reconstruction

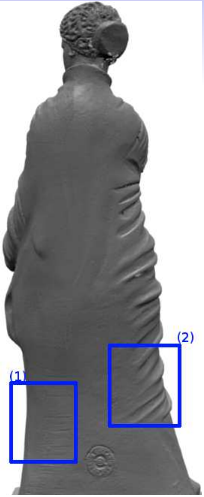
# Poisson Reconstruction of a car point set

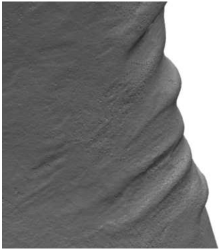
# Scale Space Meshing of a car point set

# Original Object: 20 cm high tanagra

# Coarse resolution Mesh (after projection iterations)

# Mesh obtained at a high resolution (back-projected)


(1)
(2)



# Comparison with other methods

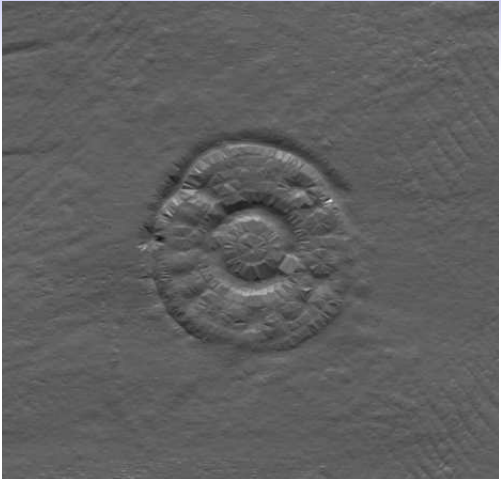


Figure: Direct meshing [BMR$^{+}$99]

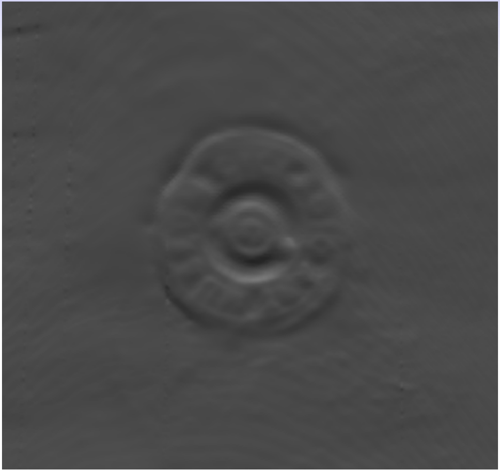


Figure: Poisson Reconstruction [KBH06]

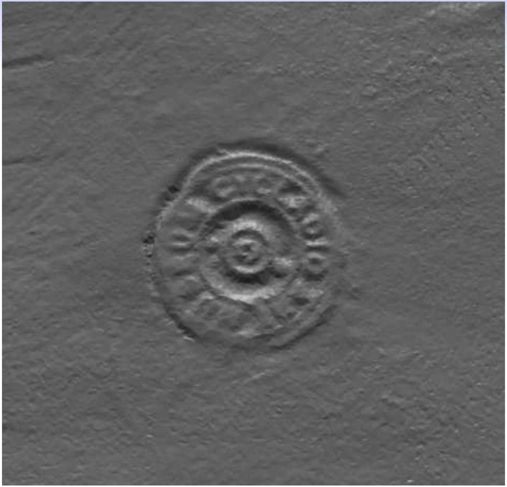


Figure: Scale Space Meshing

# Comparison between direct meshing...

# ...and scale space meshing

# Discussion: radius choice

- Every processing step relies strongly on the chosen radius

# Discussion: radius choice

- Every processing step relies strongly on the chosen radius
- Estimate the necessary radius $r$ to have approximately $n$ neighbors per point

# Discussion: radius choice

- Every processing step relies strongly on the chosen radius
- Estimate the necessary radius $r$ to have approximately $n$ neighbors per point
- $r$ should be large enough to guarantee stability but small enough to guarantee mathematical consistency

# Discussion: radius choice

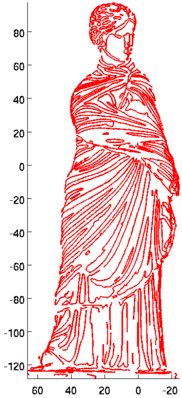
- Every processing step relies strongly on the chosen radius
- Estimate the necessary radius $r$ to have approximately $n$ neighbors per point
- $r$ should be large enough to guarantee stability but small enough to guarantee mathematical consistency
- In practice we found that $n = 20$ is enough

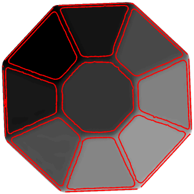
# Discussion: radius choice

- Every processing step relies strongly on the chosen radius
- Estimate the necessary radius $r$ to have approximately $n$ neighbors per point
- $r$ should be large enough to guarantee stability but small enough to guarantee mathematical consistency
- In practice we found that $n = 20$ is enough
- Sampling irregularities: in areas where the density is too low, if the distance between points is more than $2r$ no triangulation is given

The mesh allows to:

- detect curvature level lines as polylines
- detect hole borders as polylines



# Conclusion

- Thanks to a well defined curvature driven scale-space, we obtained a new way of building a robust mesh

# Conclusion

- Thanks to a well defined curvature driven scale-space, we obtained a new way of building a robust mesh
- This mesh interpolates points and preserves well details and textures

# Conclusion

- Thanks to a well defined curvature driven scale-space, we obtained a new way of building a robust mesh
- This mesh interpolates points and preserves well details and textures
- Future Work will focus on using the scale-space framework to deal with point cloud registration, crest line detection.

- Fausto Bernardini, Joshua Mittleman, Holly Rushmeier, Cláudio Silva, and Gabriel Taubin, The ball-pivoting algorithm for surface reconstruction, IEEE Transactions on Visualization and Computer Graphics 5 (1999), 349–359.
- Hugues Hoppe, Tony DeRose, Tom Duchamp, John McDonald, and Werner Stuetzle, Surface reconstruction from unorganized points, SIGGRAPH '92: Proceedings of the 19th Annual Conference on Computer Graphics and Interactive Techniques (New York, NY, USA), ACM Press, 1992, pp. 71–78.
- Michael Kazhdan, Reconstruction of solid models from oriented point sets, SGP '05: Proceedings of the Third Eurographics Symposium on Geometry processing (Aire-la-Ville, Switzerland, Switzerland), Eurographics Association, 2005, p. 73.
- Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe, Poisson surface reconstruction, SGP '06: Proceedings of the fourth Eurographics Symposium on Geometry processing (Aire-la-Ville, Switzerland, Switzerland), Eurographics Association, 2006, pp. 61–70.
- Mark Pauly, Leif P. Kobbelt, and Markus Gross, Point-based multiscale surface representation, ACM Trans. Graph. 25 (2006), no. 2, 177–193.
- Thème Sym, Frédéric Cazals, Joachim Giesen, Projet Geometrica, Unité Inria, and Sophia Antipolis, Delaunay triangulation based surface reconstruction, Effective Computational Geometry for Curves and Surfaces, Springer, 2006, pp. 231–273.