

# Modélisation surfacique

## Partie III : Reconstruction de surfaces

Ulysse Vimont

Inria, Équipe Imagine

2013-2014

- 1 Introduction
- 2 Opérations sur les nuages de points
- 3 Simplification de maillage
- 4 Inférence de voisinage
- 5 Approche différentielle
- 6 Conclusion

## 1 Introduction

- Chaîne de l'information 3D
- Définition
- Exemple
- Problématique
- Estimation
- Approches
- Références

## 2 Opérations sur les nuages de points

## 3 Simplification de maillage

## 4 Inférence de voisinage

## 5 Approche différentielle

# Chaîne de l'information 3D

- un objet réel est scanné sous plusieurs angles
- chaque résultat est pré-traité (filtrage, outliers, ...)
- les différents nuages de points sont recalés entre eux
- le résultat est post-traité (homogénéisation, filtrage, ...)
- le résultat est convertit en un modèle structuré
- cette nouvelle structure est améliorée (rebouchement des trous, ...)
- le résultat est visualisé, imprimé, animé, ...

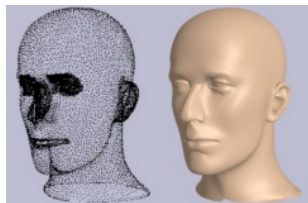
cf <http://graphics.stanford.edu/data/3Dscanrep/>.

# Définition

## Reconstruction de surface :

Passer d'un modèle de représentation d'objet ponctuel à un modèle surfacique.

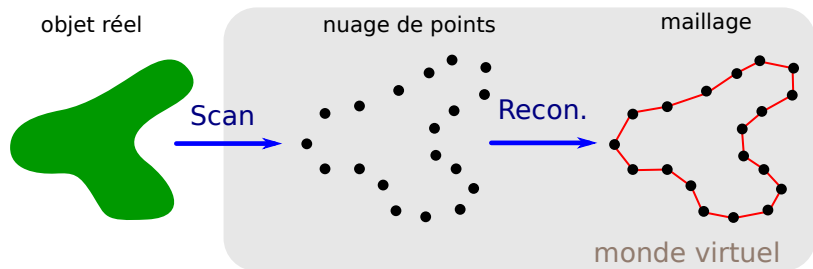
- nuage de points -> maillage, surfaces B-spline, surfaces paramétriques, surface implicite
- problème classique : mise en place d'outils standard de traitement des nuages de points



Hugues Hoppe, Surface reconstruction from unorganized points.

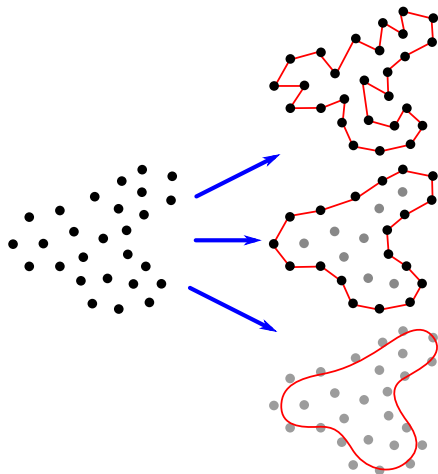
# Exemple

- virtualisation : scan d'objet réel -> maillage
- compression : perte de connectivité -> reconstruction
- remaillage : ré-échantillonnage + re-création de la connectivité
- ...



# Problématique

- création d'information :  
intégration d'*a priori*?
- robustesse : inadaptation du  
modèle ?
- conservation de l'information  
initiale ? interpolant vs.  
approximant.



# Estimation

Il existe beaucoup de méthodes de reconstruction de surface.

La qualité d'une méthode s'estime à :

- sa complexité algorithmique (capacité à reconstruire à partir de beaucoup de points)
- la qualité des résultats (capacité à reconstruire une variété sans trous)
- la résistance au bruit
- 

autre critère :

- temps réel
- adaptatif
- mettable à l'échelle (scalable)



# Approches

Différentes manières de concevoir le problème :

- simplification de maillages denses (Optimal transport theory, simplification locale, crust)
- inférence du voisinage (alpha shapes, Rotating Ball Algorithm)
- fonction de distance signée (Radial Basis Function, problème de Poisson)

Quelques sous-problèmes :

- définition des normales
- orientation des normales

# Références

Deux thèses :

- Inverse geometry : from the raw point cloud to the 3d surface : theory and algorithms, Julie Digne (UCBL)
- Surface Reconstruction from Unorganized Points, Hugues Hoppe
- ...

Coté framework :

- documentation CGAL
- documentation OpenMesh

## 1 Introduction

## 2 Opérations sur les nuages de points

- Distance de Hausdorff
- Distance à un nuage de points
- Voisinage dans un nuage de points
- Filtrage d'un nuage de points
- Calcul des normales d'un nuage de points
- Orientation des normales
- Triangulation de Delaunay

## 3 Simplification de maillage

## 4 Inférence de voisinage

## 5 Approche différentielle

# Distance de Hausdorff

## Définition : distance de Hausdorff d'un point à un ensemble

Soit  $S$  un ensemble. On définit la distance de Hausdorff du point  $A$  à  $S$  par :

$$dist(A, S) = \min(\{dist(A, s) | s \in S\})$$

- dépend d'une distance "simple" (de point à point)
- algorithme de Newton pour trouver la distance minimale

# Distance de Hausdorff

## Définition : distance de Hausdorff d'un ensemble à un autre

Soit  $S_1$  et  $S_2$  deux ensembles. On définit la distance de Hausdorff de  $S_1$  à  $S_2$  par :

$$\text{dist}(S_1, S_2) = \max(\{\text{dist}(s, S_2) \mid s \in S_1\})$$

- non symétrique ( $\text{dist}(S_1, S_2) \neq \text{dist}(S_2, S_1)$  en général)

# Distance de Hausdorff

## Définition : distance de Hausdorff symétrisée

Soit  $S_1$  et  $S_2$  deux nuages de points. On définit la distance de Hausdorff symétrisée de  $S_1$  à  $S_2$  par :

$$\text{dist}^*(S_1, S_2) = \max(\text{dist}(S_1, S_2), \text{dist}(S_2, S_1))$$

- on peut en imaginer d'autres ( $\text{dist}(S_1, S_2) + \text{dist}(S_2, S_1)$  par exemple)

# Distance à un nuage de points

Remarque : la distance d'un point à un ensemble correspond à la distance d'un ensemble ponctuel à un ensemble quelconque.

## Définition : distance de Hausdorff d'un point à un nuage de points

Soit  $P = \{p_i\}$  un nuage de points. On définit la distance de Hausdorff du point  $A$  à  $P$  par :

$$\text{dist}(A, P) = \min_i(\text{dist}(A, p_i))$$

# Distance à un nuage de points

## Définition : distance de Hausdorff entre nuages de points

Soit  $P_1 = \{p_i^1\}$  et  $P_2 = \{p_i^2\}$  deux nuages de points. On définit la distance de Hausdorff de  $P_1$  à  $P_2$  par :

$$\text{dist}(P_1, P_2) = \min_i(\text{dist}(p_i^1, P_2))$$

## Définition : distance de Hausdorff symétrisée entre nuages de points

Soit  $P_1 = \{p_i^1\}$  et  $P_2 = \{p_i^2\}$  deux nuages de points. On définit la distance de Hausdorff symétrisée de  $P_1$  à  $P_2$  par :

$$\text{dist}^*(P_1, P_2) = \max(\text{dist}(P_1, P_2), \text{dist}(P_2, P_1))$$



# Distance à un nuage de points

Application : recalage. On a deux nuages de points, et on cherche la transformation qui fait passer de l'un à l'autre

- on détermine les paramètres de la transformation qui minimise la distance entre les nuages de points
- on procède par morceaux pour recaler les nuages issus de captures sous différents angles (grâce au recouvrement)



[http://en.wikipedia.org/wiki/Point\\_set\\_registration](http://en.wikipedia.org/wiki/Point_set_registration)

## Voisinage dans un nuage de points

Soit  $P = \{p_i\}$  un nuage de points. Pour chaque  $i$  on cherche un ensemble  $v_i \subset P$  qui représente la notion de localité.

- idée 1 : On considère l'ensemble des points du nuage dans une sphère centrée autour du point considéré. Problème :
  - ▶ on ne peut être sûr que  $v_i \cap \{p_i\} \neq \emptyset$
- idée 2 : On considère un nombre donné de plus proches voisins du point considéré. Problème :
  - ▶ on ne peut être sûr de conserver la notion de localité
  - ▶ non symétrique ( $p_j \in v_i \not\Rightarrow p_i \in v_j$ )
- il faut choisir la notion correspondant à son objectif (ou faire un compromis)
- la complexité de la détermination du voisinage dépend de la structure accélératrice utilisée

# Filtrage d'un nuage de points

notion de voisinage -> notion de régularité

## Principe : lissage laplacien

Le lissage laplacien consiste à rendre un ensemble de valeurs réparties dans l'espace plus homogènes en rapprochant chaque composante de la moyenne de ses composantes voisines.

- cf. lissage laplacien pour les maillages
- différence : définition de la proximité dans l'ensemble
- résultats différents en fonction de la notion de voisinage adoptée
- problème : lissage de parties non surfaciques pures

Extension : approximation locale et projection sur un espace de régression.

# Calcul des normales d'un nuage de points

Hugues Hoppe, Surface reconstruction from unorganized points.

## Idée de base

Pour un point  $p_i$  donné :

- on détermine son voisinage  $v_i$
  - on trouve un plan  $\Pi_i$  approximant  $v_i$
  - on prend le vecteur normal  $n_i$  de  $\Pi_i$
- 
- Orientation des normales ? (ex : du plan vers le point)
  - Homogénéité de la répartition des normales ?

# Calcul des normales d'un nuage de points

Voronoi-based Variational Reconstruction of Unoriented Point Sets, P. Alliez et al., SGP 2007

## Idée de base

Pour un nuage de points  $P = \{p_i\}$  donné, on détermine son diagramme de Voronoï  $P' = \{c_i\}$  (ensemble de cellules). Ensuite, pour chaque  $p_i$  :

- on détermine les composantes principales  $\{d_j\}$  de la cellule  $c_i$  correspondante
- on prend  $n_i = d_k$  où  $k = \operatorname{argmin}_j \|d_j\|$
- la valeurs propre associée représente une mesure de confiance

- Orientation des normales ? (ex : du point vers le centre de gravité de la cellule)
- Homogénéité de la répartition des normales ?

# Orientation des normales

Idée : pour chaque point, on prend la direction moyenne des normales du voisinage (lissage laplacien des directions des normales).

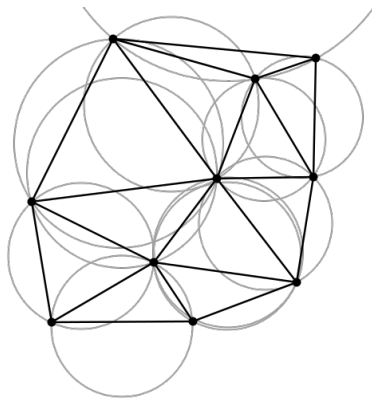
Problèmes :

- surfaces non orientables
- nuages peu denses
- zones non surfaciques

# Triangulation de Delaunay

Pour tout nuage de point, on peut trouver une triangulation telle que chaque triangle a un cercle circonscrit ne contenant aucun autre point de l'ensemble que ceux sur lesquels il s'appuie.

- notion fondamentale
- dualité avec le diagramme de Voronoï
- complexité algorithmique en  $n \times \log(n)$



- 1 Introduction
- 2 Opérations sur les nuages de points
- 3 Simplification de maillage**
  - Enveloppe convexe
  - Algorithme de Crust
  - Extension
  - Transport optimal
  - Bilan
- 4 Inférence de voisinage
- 5 Approche différentielle
- 6 Conclusion

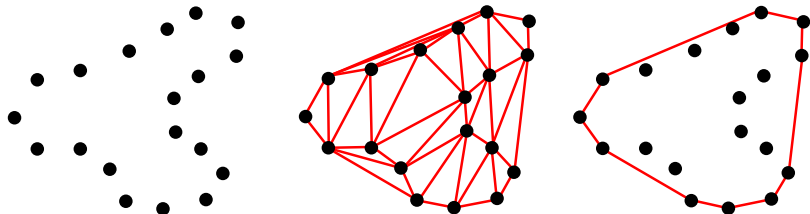


# Enveloppe convexe

Détection du contour de la triangulation de Delaunay : en 2D, on garde les arêtes adjacentes à un seul sommet

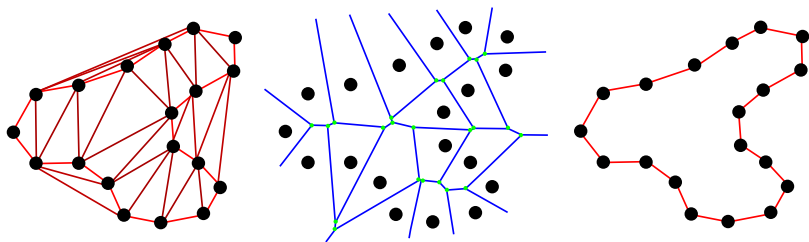
Problèmes :

- nuage volumique vs nuage surfacique
- objets non convexes
- algorithme non optimal pour extraire une enveloppe convexe (cf. Quickhull, Jarvis)



# Algorithme de Crust

- Étant donné un nuage de points  $P = \{p_i\}$ , calculer sa triangulation de Delaunay  $T = \{t_i\}$  (ensemble de triangles) et son diagramme de Voronoï  $P' = \{p'_i\}$  (ensemble de points).
- On cherche à déterminer quels sont les arêtes de  $T$  qui ne sont pas sur la surface.
- Garder les arêtes  $a$  de  $T$  pour lesquelles on peut trouver un cercle contenant  $a$  et ne contenant pas de sommet de Voronoï  $p'_i$ .



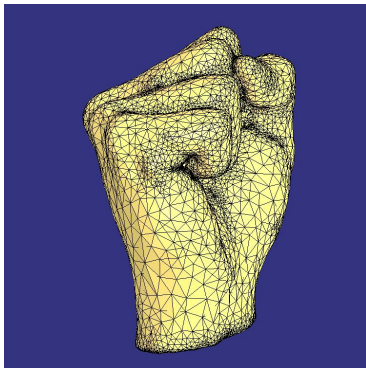
# Algorithme de Crust

Avantage :

- preuve de convergence

Inconvénient :

- qualité du maillage de sortie



## Extension

The Power Crust, Nina Amenta Sunghee, Choi Ravi and Krishna Kolluri, SMA, 2001.

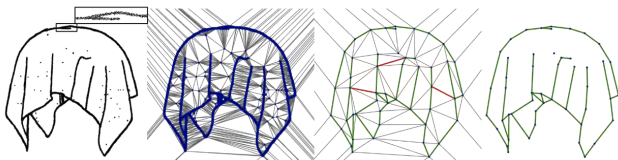


# Transport optimal

An Optimal Transport Approach to Robust Reconstruction and Simplification of 2D Shapes, Fernando de Goes, SGP, 2011

Objectif : filtrage d'un maillage reconstruit

- calcul du delaunay complet
- calcul du cout de toutes les simplifications possibles
- réalisation de la simplification la moins couteuse
- itération jusqu'à atteinte d'un seuil (nb. de pts., erreur d'approx.)



Problèmes :

- seulement en 2D
- reconstruit le bruit

Problèmes des approches par simplification de maillage :

- nécessite la construction d'un graphe intermédiaire lourd
- non scalable

cf. The Digital Michelangelo Project

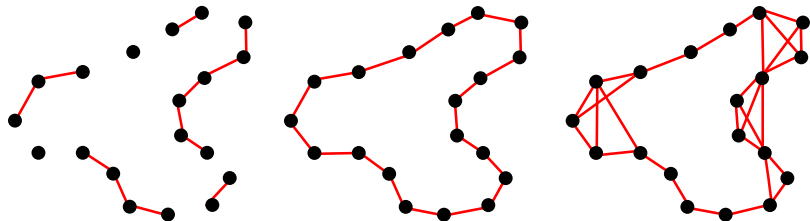


- 1 Introduction
- 2 Opérations sur les nuages de points
- 3 Simplification de maillage
- 4 Inférence de voisinage**
  - Principe
  - BPA
  - Bilan
- 5 Approche différentielle
- 6 Conclusion

# Principe

## Critère *ad hoc*

-> définition de la connectivité sous la forme de triangles s'appuyant sur des points situés dans un même voisinage.



## Problèmes :

- Définition de la localité ?
- Échantillonnage de points non homogène
- Propriétés sur la surface résultante : variété ?
- création de la triangulation de Delaunay -> scalabilité ?



# Ball Pivoting Algorithm (BPA)

The ball-pivoting algorithm for surface reconstruction, Fausto Bernardini et al., 1999

Extension des alpha-shape (mix simplification / approche locale)

Algorithme :

- on part de 3 pts sur lesquels s'appuie une sphère de rayon  $\rho$ , ils forment un triangle
- la sphère pivote autour d'une des arête, jusqu'à atteindre un nouveau triplet, qui à son tour forme un triangle

Exemple en 2D sur les slides de Julie Digne.

# Ball Pivoting Algorithm (BPA)

## Remarques :

- plus la boule est petite, plus on va créer de trous et de parties déconnectées
- plus la boule est grande, plus on va perdre de détails
- cas limite : boule de rayon  $\infty$ . Triangulation finale = enveloppe convexe.

## Extension :

- le rayon doit être choisit en fonction du pas d'échantillonnage
- rayon adaptatif en fonction du pas moyen dans un certain voisinage

Problème des approches locales :

- nécessitent une adaptation locale pour correctement reconstruire la surface
- pas d'assurance de variété topologique

- 1 Introduction
- 2 Opérations sur les nuages de points
- 3 Simplification de maillage
- 4 Inférence de voisinage
- 5 Approche différentielle**
  - RBF
  - Bilan
- 6 Conclusion

# Radial Basis Function (RBF)

Interpolation de valeurs dans l'espace (scattered data interpolation) :

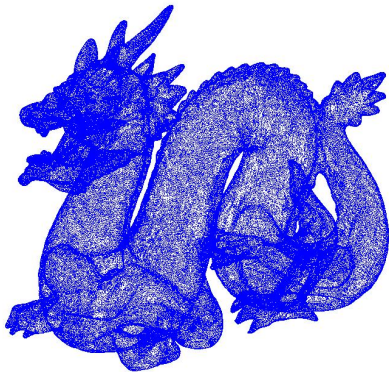
- Soit  $N$  point  $p_i$  de l'espace affectés d'autant de valeurs  $w_i$ . On cherche une fonction continue de l'espace  $f$  telle que  $\forall i, f(p_i) = w_i$ .
- On considère une fonction radiale  $r$  (par exemple  $r(x) = \|x\|$ ), qu'on va utiliser comme base pour exprimer  $f$  :  $f(x) = \sum f_i r(x - p_i)$ .
- On note  $\phi_{i,j} = r(p_i - p_j)$ , et  $\Phi = (\phi_{i,j})$  (matrice symétrique)
- Le problème s'exprime par  $\Phi \times F = W$  où :
  - ▶  $F = (f_i)$
  - ▶  $W = (w_i)$

démo octave

# Radial Basis Function (RBF)

Reconstruction and Representation of 3D Objects with Radial Basis Functions, J. C. Carr et al., SIGGRAPH 2001.

On peut utiliser les RBF pour reconstruire une surface :



# Radial Basis Function (RBF)

- on détermine une fonction implicite qui va définir notre surface interpolante
- contrainte  $f(p_i) = 0$  insuffisante (on obtient la fonction nulle)
- pour chaque point, dans la direction de sa normale, on définit deux nouveaux points de contrainte :
  - ▶ un à l'intérieur avec un poids positif
  - ▶ un à l'extérieur avec un poids négatif

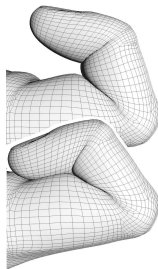
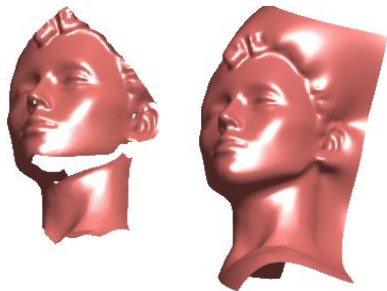
## Remarques :

- autre solution : prendre en compte le gradient dans la résolution
- on résout un problème de Poisson
- on peut rajouter un terme polynomiale à l'expression de  $f$  pour minimiser la courbure
- on obtient un maillage par un algo type marching cube.

# Radial Basis Function (RBF)

Autre utilisation :

- rebouchage de trous (interpolation vs extrapolation)
- Implicit Skinning : Real-Time Skin Deformation with Contact Modeling





Inconvénient :

- nécessite la résolution d'un système linéaire
- non scalable et potentiellement lent

# Conclusion

Il existe bien d'autres méthodes non présentées ici, par exemple à SIGGRAPH 2013 ([s2013.siggraph.org/attendees/technical-papers/session/surface-reconstruction](http://s2013.siggraph.org/attendees/technical-papers/session/surface-reconstruction)) :

- Screened Poisson-Surface Reconstruction, Kazhdan and Hoppe
- A Benchmark for Surface Reconstruction, Berger, Levine, Nonato, Taubin and Silva
- Dense Scene Reconstruction With Points of Interest, Zhou and Koltun
- Scalable Real-Time Volumetric Surface Reconstruction, Jiawen Chen, Bautembach and Izadi

# Conclusion

Dant tout les cas, il faut faire un compromis entre :

- la rapidité
- la robustesse
- la scalabilité

Fin !