

TP - Reconstruction de formes 3D

CPE - Julie Digne

19 Octobre 2022

Dans ce TP nous verrons comment reconstruire un maillage 3D à partir d'un nuage de points en utilisant deux types de méthode :

- La méthode "historique" [Hoppe 92]
- Une méthode récente de Représentation Neuronale Implicite [Gropp 2020]

1 Librairies - logiciels

Nous visualiserons les données sous blender (maillages) ou meshlab (nuages de points). La programmation se fait en python, en utilisant les librairies suivantes

- Numpy
- PyMCubes (implémentation des marching cubes)
- Matplotlib (pour visualiser l'évolution des fonctions de coût)
- scipy (pour le Kdtree)
- Pytorch (pour la partie neuronale)

2 Méthode classique

La méthode *Surface reconstruction from Unorganized Point Sets* [Hoppe 1992] est une méthode historique de reconstruction de surface. Elle consiste à prendre un nuage de points orienté (p_i, n_i) , et à estimer en tout une fonction distance signée en tout point q comme : $f(q) = \pm \min_i \|p_i - q\|$

Le signe est donné par la valeur du produit scalaire $\langle q - p_i, n_i \rangle$.

NB : La méthode originale part d'un nuage non orienté et en calcule l'orientation, ici pour simplifier les données ont des normales correctement orientées précalculées.

1. Lire le nuage de points (np.loadtxt)
2. Créer un kdtree encodant ce nuage de points (spatial.KDTree dans scipy.spatial)
3. Créer une grille de points (np.mgrid)
4. Calculer les valeurs de distance signée en tout point de cette grille
5. Extraire le niveau 0 (marching_cubes de mcubes)
6. Sauver le maillage (export_obj de mcubes)

3 Méthode par représentation implicite neuronale

La seconde méthode est inspirée des travaux de [Gropp et al. 2020]. Il s'agit d'entraîner un réseau de neurones qui, pour tout point q de l'espace ambiant, retourne $u(q)$, la fonction distance signée à la forme.

1. Créer un MLP de n_l couches avec n_n neurones par couche avec des activations ReLU sauf pour la dernière (activation linéaire)
2. Optimiser le réseau pour la forme, pour cela on utilise deux losses :

- Une loss qui force les points de la surface à avoir une distance 0 et à avoir des gradients alignés avec les normales à la surface.
 - Une loss qui force tous les points de l'espace ambiant à avoir $\|\nabla u\| = 1$
3. Une fois le réseau entraîné, calculer les valeurs de distance signée en tout point de cette grille
 4. Extraire le niveau 0 (marching_cubes de mcubes)
 5. Sauver le maillage (export_obj de mcubes)