

Reconstruction 3D et Géométrie dans l'espace

Par **Keyvan Beroukhim & Sabrina Chaouche**
Encadrants **Nicolas Baskiotis & Vincent Guigue**



Plan

1. Introduction: Le projet
2. La reconstruction 3D
3. Etat de l'art de la reconstruction 3D
4. Prise en main des données
5. Méthode Atlas-Net: Architecture et Fonction de coût
6. Notre implémentation et nos résultats
7. Affichage des faces reconstruites
8. Conclusion



Introduction

- Travailler sur la reconstruction 3D à partir des images 2D avec des nuages de points comme supervision.
- Mettre en place une architecture capable d'analyser les images fournies en entrée et d'estimer la forme des objets en sortie.
- Basé sur les travaux de Thibault Groueix chez Adobe Research.

Domaines d'application de la reconstruction 3D



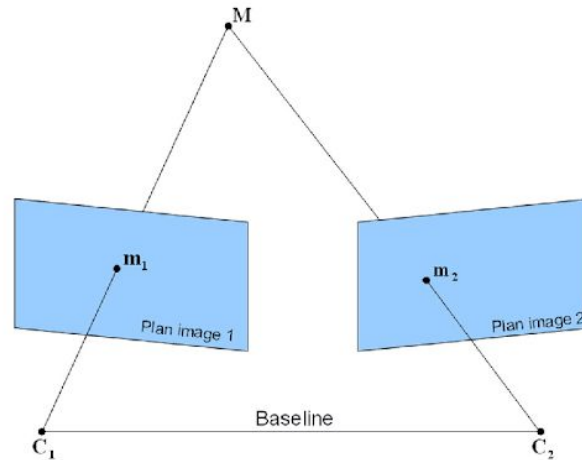
Reconstruction 3D d'un visage



Réalité augmentée

Etat de l'art en reconstruction 3D

La stéréovision - Reconstruction 3D à partir de plusieurs images



Le principe de la stéréovision

Etat de l'art en reconstruction 3D

Point Net

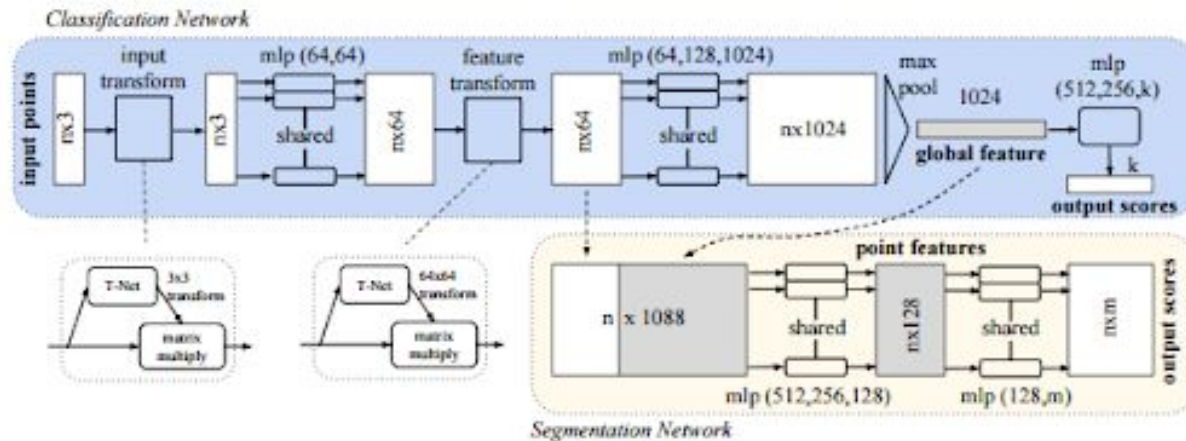
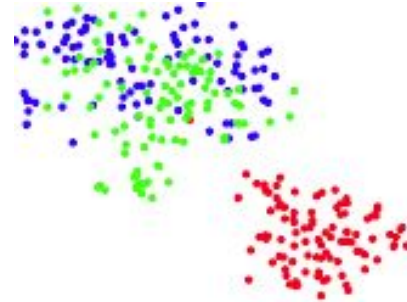


Figure illustrant l'architecture de Point Net

Prise en main des données - t-SNE

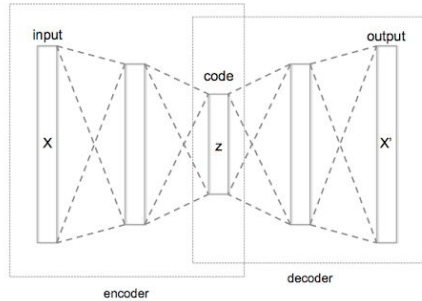


Visualisation du résultat du
t-SNE pour les 13 classes

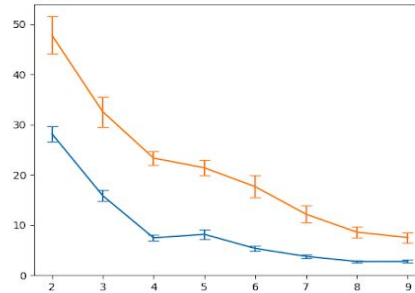


Exhibition de 2 classes
confondues et une classe
distincte

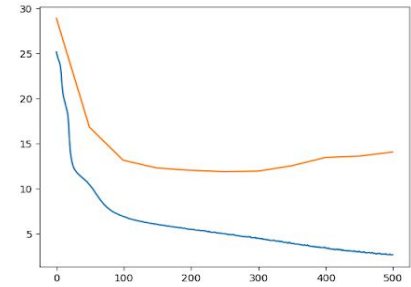
Prise en main des données - Auto Encodeur



Exemple d'architecture
d'un auto-encodeur

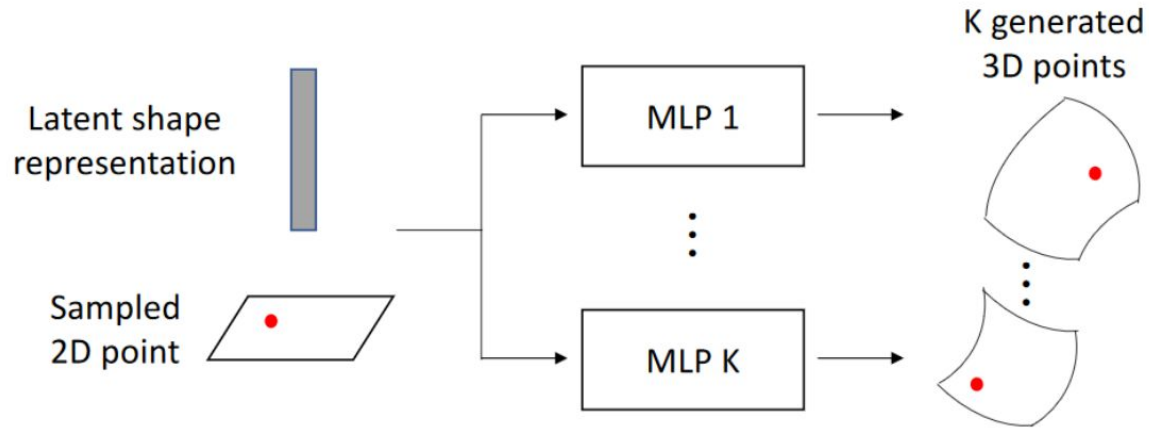


Erreur en fonction de la
taille du vecteur latent



Erreur en fonction du
nombre d'epochs

Atlas Net - Architecture



Architecture de Atlas Net (source: article de Groueix)



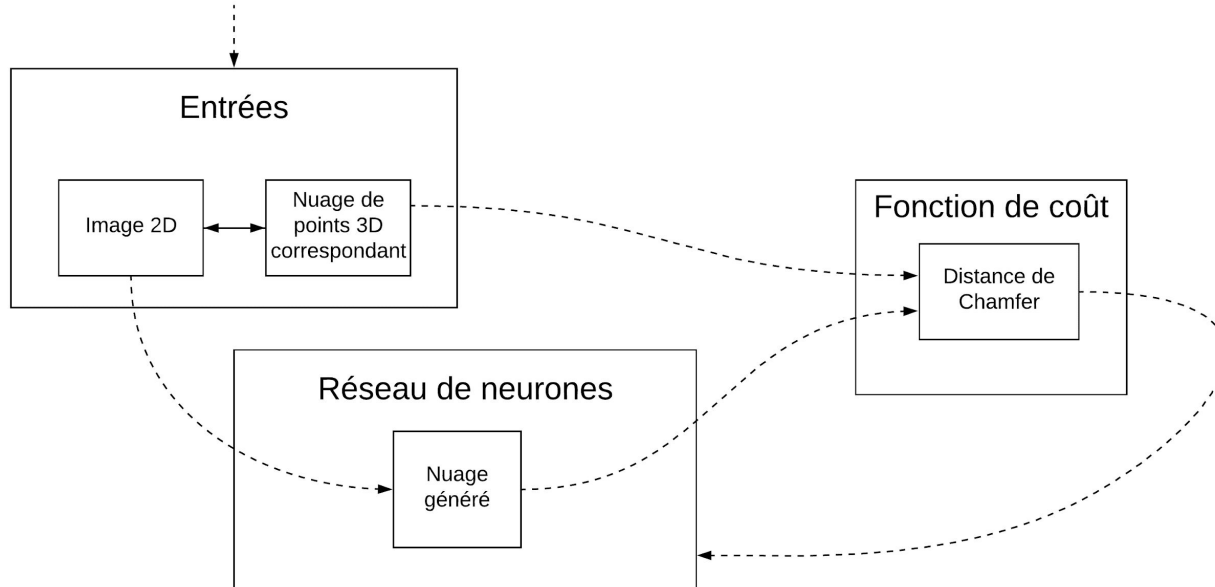
Atlas Net - Fonction de coût

Distance de Chamfer

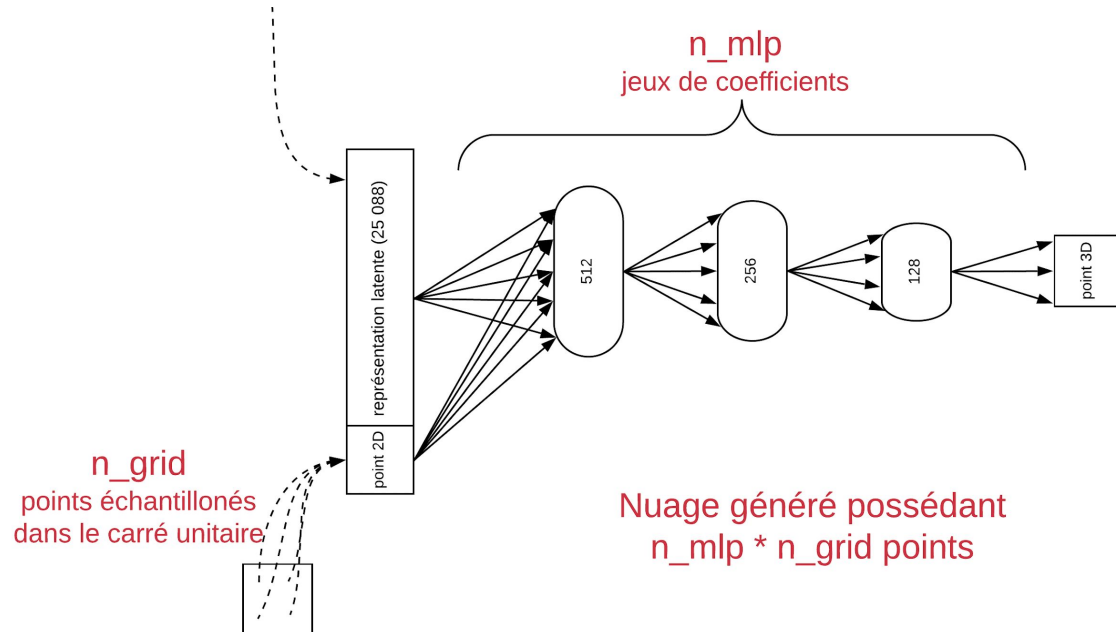
$$CD(Y, S^*) = \sum \min_{s \in S^*} \|\mathbf{y} - \mathbf{s}\|^2 + \sum \min_{y \in Y} \|\mathbf{y} - \mathbf{s}\|^2$$

- Minimise pour chaque point de S^* , la distance au point généré y le plus proche \Rightarrow les points de S^* doivent être atteints par la forme générée.
- Minimise la distance pour chaque point généré au point de S^* le plus proche \Rightarrow les points générés ne doivent pas dépasser la forme originale.

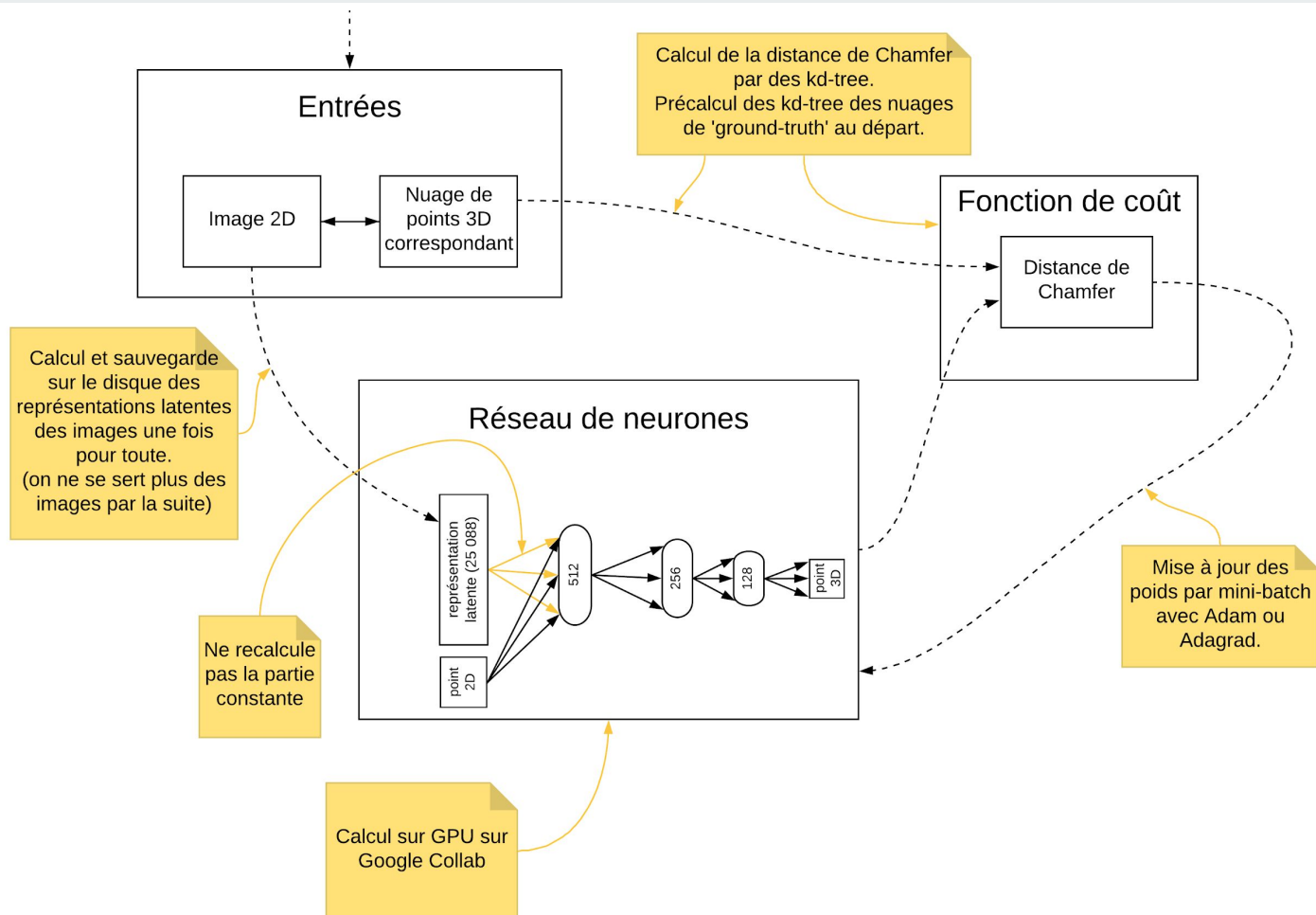
Boucle d'apprentissage



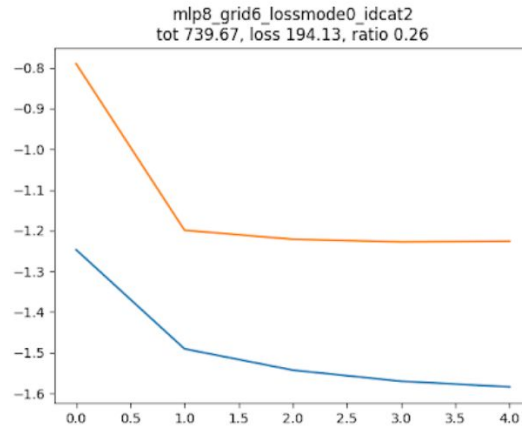
Architecture du réseau de neurones



Améliorations



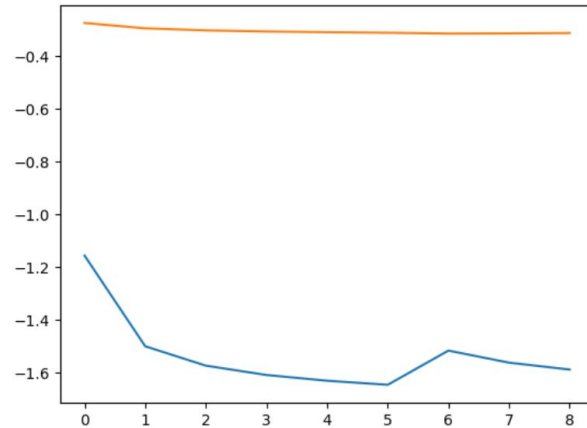
Entraînement du réseau



Loss en fonction de l'époch

- train (bleu) converge
- test (orange) décroît au début mais ne converge pas vers 0
- même scénario quel que soit le type d'objet

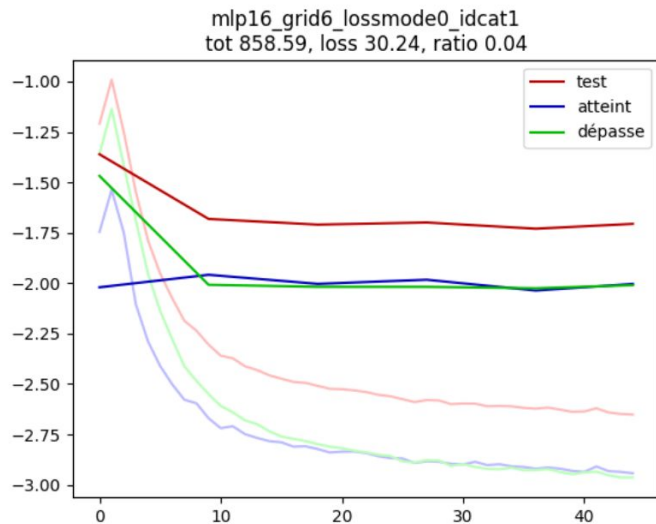
Entraînement du réseau



Changement de loss en train à l'epoch 6

- aucun effet sur la loss en test

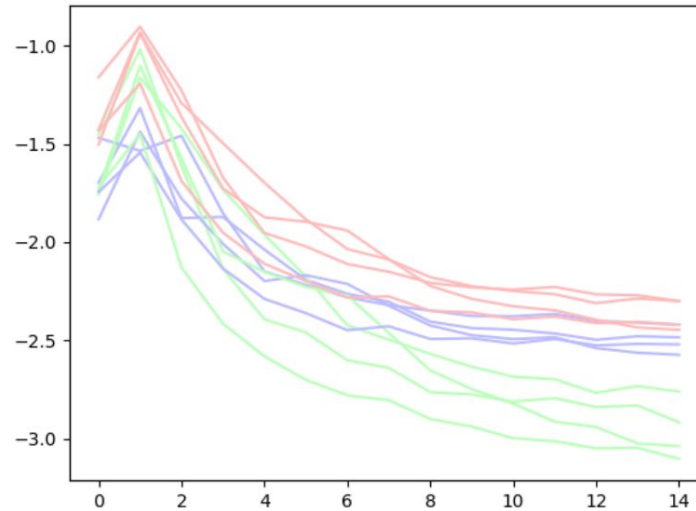
Entraînement du réseau



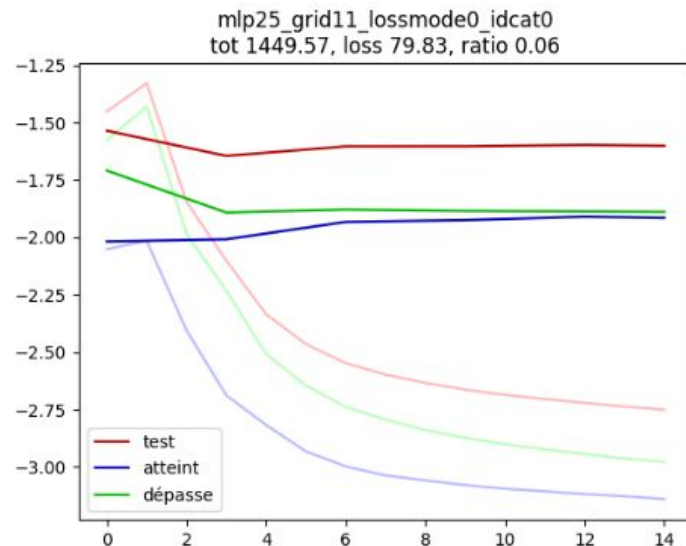
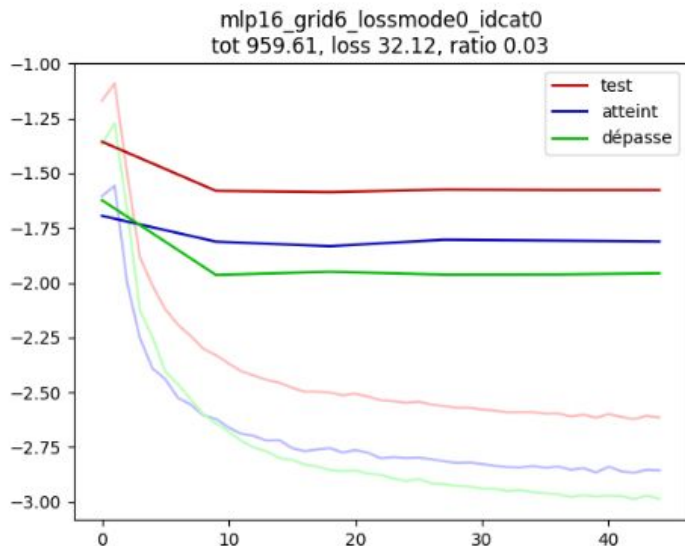
- 1er terme de la loss (bleu)
- 2ème terme (vert)
- total (rouge)
- initialisation des poids avec la méthode “Xavier”
- échelle logarithmique

Les courbes indiquent les loss en train (clair) et test (foncé) en détail

Entraînement du réseau



Loss sur 4 nuages menant à une reconstruction visuellement incorrecte, les erreurs sont similaires.



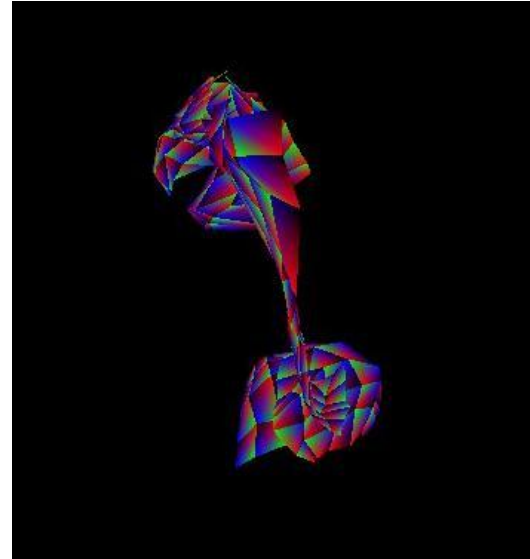
Nuages générés de taille $16 * 6 * 6 = 576$ et des nuages de supervision de taille 300 (à gauche) de taille $25 * 11 * 11 = 3025$ et une supervision de taille 3000

Tentatives:

- Taille des nuages générés et de supervision
- Rééchantillonnage de la grille et/ou des nuages
- Poids des termes de la loss
- Loss définie par une moyenne ou une somme
- Avec ou sans mini-batch

Reconstruction des faces

- Utiliser la grille de génération des nuages de points pour trouver les sommets connectés.
- Pas besoin de connecter les patches
- Rendu en OpenGL





Conclusion

- Découverte de la reconstruction 3D
- Découverte de PyTorch et des réseaux de neurones
- Résolution de problèmes de complexité calculatoire de manière logicielle et matérielle
- Bloqués face au problème du fonctionnement en 'boîte noire' des réseaux de neurones



Merci pour votre attention