

Compte rendu PLDAC géométrie reconstruction 3D
BEROUKHIM CHAOUCHE
encadrants Baskiotis Guigue
date

Intro

1) Article

bases : PyTorch, CNN, VGG, ImageNet, autres trucs mentionnés par Groueix
exécution du classifieur

2) tSNE

CNN => autoencoder ?

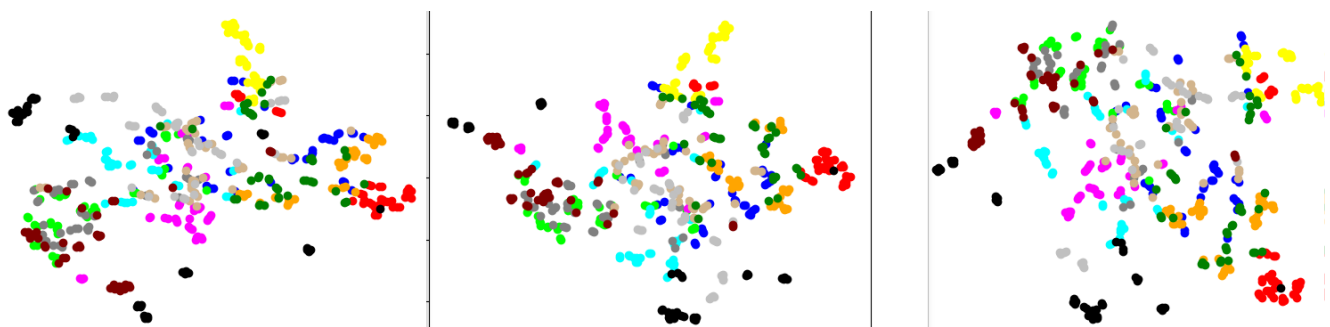
VGG est un réseau de neurones composé de 36 couches consécutives. Les 30 premières couches résument l'entrée en un plus petit vecteur, c'est la partie « feature extraction ». Les couches suivantes réalisent l'étape de classification à proprement parler.

La partie classification de VGG ne nous intéresse pas, nous nous servons du réseau comme extracteur de caractéristique en stoppant l'exécution du réseau de neurone à une couche intermédiaire (on récupère un vecteur latent).

VGG prend en entrée des images RGB 224*224, la dimension de l'entrée est donc $224*224*3 = 150\,528$. Le vecteur latent récupéré est de dimension 25 088.

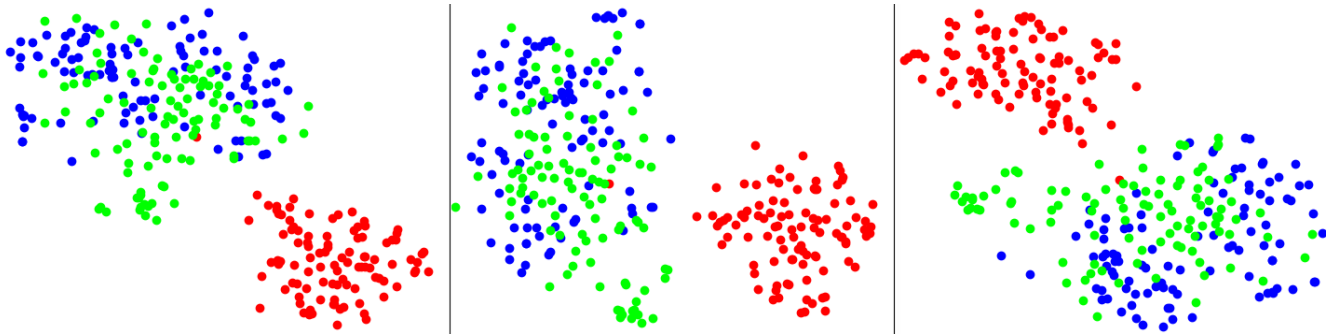
L'algorithme t-SNE (T-distributed Stochastic Neighbor Embedding) est un algorithme servant à visualiser des données de grande dimension. Il prend en entrée des points dans un espace de grande dimension et les ramène dans un espace à 2 ou 3 dimensions tout en tentant de préserver les distances entre les points.

La dimension des vecteurs latents étant trop élevée, nous commençons par réduire le nombre de dimensions par PCA (Principal Component Analysis) afin de réduire le bruit (comme demandé dans la documentation de sklearn.tSNE).



Résultat avec les 13 classes

Les différentes classes d'objet se regroupent à peu près en cluster, cela démontre la capacité du réseau de neurone à résumer les informations dans un espace de plus petite taille que celle de l'entrée.



Exhibition de deux classes confondues et une distincte
(rouge: avion – bleu: meuble – vert: enceintes)

3) Classifieur

Nous avons entraîné un réseau simple prenant en entrée les vecteurs latents et classifiant en deux classes. Les performances obtenues sont :

| | | | |
|--------------------|---|-------------|------------|
| meuble – avion | : | train 100 % | test 100 % |
| meuble – enceintes | : | train 78 % | test 66 % |

4) Distances

Chamfer minimise

pour chaque P et MLP la distance au Q le plus proche.

Les MLP doivent bien atteindre un Q

pour chaque Q la distance au P MLP le plus proche

Les Q doivent bien être atteint par un MLP

5) Explication article

A partir de nuages de points, on génère des images 2D. A partir de ces images 2D, on peut entraîner un réseau pour deux tâches distinctes :

- reconstruire le nuage de point (c'est ce que fait PointNet)
- classifier les images (c'est ce que font ResNet et VGG)

Bien que notre but soit de reconstruire un nuage de point, on pourra essayer de reconstruire les nuages de points à partir des vecteurs latents issus des réseaux de neurones classifieurs.