

TP : Apprentissage par transfert (Transfer Learning) pour la segmentation d'images

La tâche de classification pour les images est relativement simple par rapport à la segmentation d'images. L'objectif de ce TP est de construire des modèles de segmentation d'images en utilisant l'apprentissage par transfert pour construire un mask de la vérité du terrain.

NB1 : Assurez-vous de documenter votre code de manière précise, d'inclure des graphiques illustrant les performances des modèles, et d'interpréter de manière approfondie les résultats obtenus.

NB2 : La base de données à utiliser est **COCO**: <https://cocodataset.org/#home>

Exercice 1: Implémentation du Unet

Référence pour l'architecture du Unet : <https://arxiv.org/pdf/1505.04597v1.pdf>

1. Implémenter l'architecture de Unet en utilisant les frameworks keras/TensorFlow ou PyTorch. Ensuite, exécuter l'entraînement et présenter les résultats obtenus.
2. Améliorer l'architecture en ajoutant des blocs d'attention
3. Améliorer l'architecture en utilisant un backbone Resnet

Exercice 2: Implémentation du Deeplabv3+

Référence pour l'architecture du DeeplabV3+ : <https://arxiv.org/pdf/1802.02611.pdf>

1. Implémenter l'architecture de Deeplabv3+ (backbone=Xception, outputStride=16) en utilisant les frameworks keras/TensorFlow ou PyTorch. Ensuite, exécuter l'entraînement et présenter les résultats obtenus.
2. Améliorer l'architecture en modifiant le 'backbone' et 'output Stride'
3. Améliorer l'architecture en modifiant le 'output stride'

Exercice 3: GAN - Générations des données

En utilisant le même jeu des données COCO, utilisez l'architecture GAN pour générer de nouvelles images.

Référence pour l'architecture du GAN : <https://arxiv.org/pdf/1406.2661.pdf>

1. Quels sont les avantages spécifiques des modèles GAN ?
2. Quelles sont les implémentations de ces architectures?
3. Améliorer l'architecture avec l'attention