

Plan Prévisionnel - Projet 9 : SegFormer pour Segmentation Cityscapes

Étudiant : Didier DRACHE

Date : Fin juillet 2025

Objectif : Preuve de concept avec amélioration performance vs VGG16-UNet (P8)

Dataset retenu

Le dataset **Cityscapes** sera réutilisé pour assurer une comparaison rigoureuse avec la baseline du Projet 8. Ce dataset de référence pour la segmentation sémantique urbaine contient 5 000 images haute résolution avec annotations précises de 34 classes, regroupées en 8 catégories principales. Le choix stratégique de Cityscapes permet une évaluation directe des gains tout en conservant l'infrastructure de données existante (IoU baseline = 0.622).

Modèle envisagé

SegFormer (NeurIPS 2021) est choisi pour son architecture révolutionnaire de Vision Transformers hiérarchiques. Arguments justifiant ce choix : (1) **Performance prouvée** - 84% mIoU sur Cityscapes, soit +35% vs baseline ; (2) **Architecture innovante** - encodeur ViT multi-échelles sans positional encoding ; (3) **Efficacité edge** - modèle B2 (~90MB) compatible Azure F1 ; (4) **Innovation Mix-FFN** - combine attention globale et convolutions locales.

Objectif : Segmentation sémantique haute performance. **Contexte** : Véhicules autonomes nécessitant temps réel et >80% mIoU.

Références bibliographiques

1. Article de recherche principal

Xie, E. et al. (2021). “*SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers*”. NeurIPS 2021. ArXiv: arxiv.org/abs/2105.15203

2. Guide d'implémentation

Hugging Face Blog (2022). “*Fine-tune SegFormer on a custom dataset*”. huggingface.co/blog/fine-tune-segformer

3. Benchmark officiel Cityscapes

Cityscapes Benchmark Suite (2024). “*Semantic Segmentation Leaderboard*”. www.cityscapes-dataset.com/benchmarks

Démarche de test et preuve de concept

Baseline : Modèle VGG16-UNet du P8 (IoU=62.2%, temps=3.2-8.5s sur Azure F1).

Méthode SegFormer : Transfer learning depuis ImageNet, fine-tuning Cityscapes 8 classes, optimisation ONNX pour Azure F1.

Plan d'implémentation : Phase 1 - Setup Hugging Face + adaptation architecture (3j) ; Phase 2 - Entraînement et validation métriques (5j) ; Phase 3 - API Flask + Dashboard Streamlit + déploiement Azure (3j).

Objectif POC : Démontrer amélioration >10% IoU avec réduction significative du temps d'inférence et dashboard cloud fonctionnel.