

Plan Prévisionnel - Preuve de Concept

Segmentation Sémantique

Projet : Amélioration de la segmentation sémantique temps réel avec SegFormer

Dataset Retenu

Cityscapes : Dataset de référence pour la segmentation sémantique de scènes urbaines comprenant 2975 images haute résolution (2048×1024) annotées pixel par pixel. Les images sont capturées dans **50 villes allemandes différentes** avec **34 classes principales** que nous regroupons en **8 catégories pertinentes** (flat, vehicle, human, construction, object, nature, sky, void).

Particularité méthodologique : Les images seront testées aux résolutions 512×512 et 768×768 avec **déformation du ratio d'origine** (passage de 2:1 à 1:1) pour optimiser les contraintes computationnelles GPU T4. Cette modification permettra d'analyser l'impact du ratio d'image sur les différentes architectures.

Modèle Envisagé

Algorithme : SegFormer (Focus Temps Réel)

Justifications du choix :

SegFormer (Xie et al., 2021) représente une approche Transformer vision-optimisée pour la segmentation sémantique. **L'analyse du benchmark Cityscapes révèle que SegFormer-B0 égale les performances de DeepLabv3+ (75.3% vs 75.2% mIoU) tout en fonctionnant à résolution réduite** (côté court 768px vs résolution pleine). Cette efficacité computationnelle supérieure suggère un potentiel d'amélioration face à notre baseline FPN, DeepLabv3+ étant théoriquement plus performant que FPN.

Objectif et hypothèses de recherche :

1. **SegFormer-B0** devrait surpasser FPN+EfficientNetB0 (78.67% mIoU baseline) grâce à l'attention globale native
2. **SegFormer-B1** permettra d'explorer le scaling architectural (non documenté dans le papier original)
3. **Impact du ratio d'image** : Analyser comment la déformation 2:1→1:1 affecte les architectures CNN vs Transformer

Contraintes et adaptations :

- Migration framework obligatoire : TensorFlow/Keras → PyTorch pour accès aux modèles SegFormer récents
- Résolutions contraintes par GPU T4 : 512×512 et 768×768 maximum
- Protocole différent du papier original : images carrées vs rectangulaires natives

Références Bibliographiques

1. **Xie, E., Wang, W., Yu, Z., et al.** (2021). *SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers*. NeurIPS 2021. [Architecture Transformer de référence]
2. **Chen, L.C., Zhu, Y., Papandreou, G., et al.** (2018). *Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation*. ECCV 2018. [Baseline comparative DeepLabv3+]
3. **Cordts, M., Omran, M., Ramos, S., et al.** (2016). *The Cityscapes Dataset for Semantic Urban Scene Understanding*. CVPR 2016. [Dataset de référence]

Explication de la Démarche de Test

Méthode Baseline

FPN + EfficientNetB0/ResNet34 : Réentraînement des modèles du projet précédent aux nouvelles résolutions 512×512 et 768×768 avec déformation du ratio d'image. Performance baseline FPN+EfficientNetB0 : 78.67% mIoU (baseline de référence).

Méthode de Test SegFormer

Évaluation comparative temps réel :

1. **SegFormer-B0** : Modèle compact ciblé (3.7M paramètres) pour validation du concept temps réel
2. **SegFormer-B1** : Exploration scaling architectural (13.7M paramètres) non documenté dans le papier
3. **Comparaison 512×512 vs 768×768** : Trade-off résolution/performance sur images déformées
4. **Analyse différentielle** : Impact du ratio d'image 1:1 vs 2:1 natif Cityscapes

Métriques d'évaluation :

- Mean IoU (métrique principale) et accuracy pixel-wise
- Performance par classe sémantique (focus classes critiques : Human, Vehicle)
- Temps d'inférence CPU/GPU pour validation déploiement pratique

Preuve de Concept

Interface Streamlit déployée sur serveur NAS personnel (CPU Intel N100) permettant :

- Upload d'images urbaines avec **inférence comparative multi-modèles**
- **Visualisation côte-à-côte** : FPN+ResNet34 vs SegFormer-B1 (meilleurs représentants)
- Affichage temps d'inférence réel et métriques de performance détaillées
- **Analyse exploratoire dataset** : Distribution classes, statistiques pixel-wise interactives

Hypothèses de validation :

1. **SegFormer-B1 > FPN baselines** malgré contraintes ratio d'image
2. **SegFormer-B0 vs FPN** : Révélation possible de limitations Transformer sur images déformées
3. **Performances générales** potentiellement supérieures au papier original grâce à l'optimisation ratio 1:1

Code réutilisé : Framework `segmentation_models_pytorch` pour les implémentations. **Application originale** : protocole d'évaluation sur images carrées et analyse comparative CNN vs Transformer sous contraintes géométriques spécifiques.

Note méthodologique : Cette approche permet d'analyser les trade-offs pratiques architecture/performance dans un contexte de contraintes réelles (hardware limité, déformation d'images), offrant une perspective complémentaire aux évaluations académiques standard.