Note Méthodologique : Preuve de concept SegFormer vs FPN

Dataset retenu

Cityscapes Dataset : Référence en Segmentation Urbaine

Le **Cityscapes Dataset** constitue la référence internationale pour l'évaluation des algorithmes de segmentation sémantique en environnement urbain. Développé par l'Université de Tübingen et Mercedes-Benz, ce dataset comprend **25,000 images** haute résolution (2048×1024 pixels) capturées dans 50 villes allemandes et suisses.

Composition du dataset :

- **Train** : 2,975 images finement annotées
- **Validation** : 500 images avec annotations de référence
- **Test** : 1,525 images (annotations non-publiques)
- **Coarse** : 20,000 images avec annotations grossières

Architecture de classes hiérarchique :
Le dataset organise 34 classes détaillées en **8 méta-classes** pour la segmentation :

Méta-classe	Classes incluses	Importance
Flat	Route, trottoir, parkings	Navigation de base
Human	Piéton, cycliste	**Sécurité critique**
Vehicle	Voiture, camion, bus, train	Obstacles dynamiques
Construction	Bâtiment, mur, clôture, pont	Structure urbaine
Object	Poteau, feu, panneau	Signalétique
Nature	Végétation, terrain	Environnement
Sky	Ciel	Contexte
Void	Zones non-définies	Exclusion

Spécificités techniques :

- **Résolutions d'acquisition** : Données natives 2048×1024 pixels
- **Conditions d'acquisition** : Conditions météo bonnes à moyennes, pas de conditions adverses (pluie, neige)
- **Diversité géographique** : Villes allemandes et suisses (architectures variées)
- **Annotations pixel-perfect** : Masques de segmentation précis au pixel près

Ce dataset permet d'évaluer la capacité des modèles à comprendre les scènes urbaines complexes, élément fondamental pour les applications de conduite autonome et de surveillance urbaine intelligente.

Les concepts de l'algorithme récent

SegFormer : Architecture Transformer Vision-Optimisée

SegFormer représente une avancée majeure dans la segmentation sémantique en introduisant une architecture **Transformer** spécifiquement conçue pour la compréhension visuelle dense, contrairement aux Transformers textuels originaux.

Innovations Architecturales Clés

1. Encodeur Hiérarchique Multi-Échelles

L'encodeur SegFormer **abandonne les encodages positionnels** classiques au profit d'une architecture hiérarchique qui capture naturellement les relations spatiales :

$$\text{Feature Maps} = \{F_1, F_2, F_3, F_4\}$$

avec résolutions $\{\frac{H}{4}, \frac{H}{8}, \frac{H}{16}, \frac{H}{32}\}$

Efficient Self-Attention :
Contrairement à l'attention standard $\mathcal{O}(N^2)$, SegFormer utilise une **attention réduite** :

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

où K est sous-échantillonnée par un facteur R pour réduire la complexité :

$$\text{Complexité} = \mathcal{O}\left(\frac{N^2}{R}\right)$$

2. Mix-FFN : Convolution dans le Transformer

Innovation majeure : intégration de **convolutions 3×3** dans les couches Feed-Forward :

$$\text{Mix-FFN}(x) = \text{MLP}(\text{GELU}(\text{Conv}_{3\times3}(\text{MLP}(x)) + x))$$

Cette hybridation **CNN-Transformer** permet :

- **Capture des patterns locaux** (force des CNN)
- **Compréhension globale** (force des Transformers)
- **Élimination des encodages positionnels** : Contrairement aux ViT classiques qui utilisent des encodages positionnels, SegFormer s'en affranchit totalement

3. Décodeur MLP Unifié

SegFormer simplifie drastiquement le décodeur avec un **MLP léger** :

$$\text{Seg} = \text{MLP}\left(\text{Concat}[\text{Upsample}(F_1), F_2, F_3, F_4]\right)$$

Avantages :

- **Élimination des décodeurs complexes** (PSP, ASPP)
- **Réduction paramètres** : Focus sur l'encodeur
- **Inférence accélérée** : Décodeur 10× plus rapide

Comparaison Conceptuelle : SegFormer vs CNN

Aspect	CNN (FPN)	SegFormer
Réceptive Field	Locale → Globale (graduelle)	**Globale dès L1**
Relations Spatiales	Convolutions fixes	**Attention adaptative**
Multi-échelles	Feature Pyramid complexe	**Hiérarchie naturelle**
Paramètres	Nombreux (convolutions)	**Optimisés (attention)**
Contexte	Limité (kernels 3×3-5×5)	**Illimité (global)**
Encodage positionnel	Aucun (spatial intrinsèque)	**Aucun (Mix-FFN)**

Mécanisme d'Attention Global

Force distinctive : Chaque pixel "voit" tous les autres pixels de l'image simultanément :

$$\text{Output}_i = \sum_{j=1}^{N} \alpha_{ij} \cdot V_j$$

où α_{ij} représente l'**importance du pixel j pour classifier le pixel i**.

Impact :

- **Classes complexes** (Human, Vehicle) : Gains significatifs vs CNN
- **Cohérence spatiale** : Réduction artefacts de segmentation
- **Robustesse occlusion** : Reconstruction contextuelle

Cette architecture explique la **supériorité empirique** de SegFormer sur les tâches nécessitant une compréhension contextuelle étendue, particulièrement visible sur Cityscapes.

La modélisation

Méthodologie Expérimentale

Architecture comparative :
Nous comparons SegFormer à **FPN** (Feature Pyramid Network), architecture CNN mature représentative des approches

traditionnelles, avec deux encodeurs testés : **EfficientNet-B0** et **ResNet34**.

Configuration expérimentale :

- **Plateforme** : Google Colab Pro avec GPU T4 (15Go VRAM)
- **Framework** : PyTorch + Segmentation Models PyTorch
- **Contraintes** : Plan payant Google avec unités de calcul limitées

Preprocessing et Modèles Pré-entraînés

Résolutions testées : 512×512 et 768×768 pixels

Modèles SegFormer (pré-entraînés ADE20K) :

- **SegFormer-B0** : `smp-hub/segformer-b0-512x512-ade-160k`
- **SegFormer-B1** : `smp-hub/segformer-b1-512x512-ade-160k`

Modèles FPN (pré-entraînés ImageNet) :

- **FPN+ResNet34** : `encoder_weights="imagenet"`
- **FPN+EfficientNet-B0** : `encoder_weights="imagenet"`

Métrique d'Évaluation

Mean Intersection over Union (mIoU) - Métrique principale :

L'IoU mesure la précision de segmentation par classe :

$$\text{IoU}_{\text{classe}} = \frac{|\text{Prédiction} \cap \text{Ground Truth}|}{|\text{Prédiction} \cup \text{Ground Truth}|}$$

Mean IoU (mIoU) : Moyenne des IoU sur toutes les classes

$$\text{mIoU} = \frac{1}{N} \sum_{i=1}^{N} \text{IoU}_i$$

Métrique complémentaire :

- **IoU par classe (IoUc)** : Performance détaillée par méta-classe

Démarche d'Optimisation

Configuration SegFormer :

- **Loss Function** : Jaccard Loss seule
- **Optimiseur** : AdamW (lr=1e-4, weight_decay=1e-4)
- **Scheduler** : Cosine Annealing (min_lr=1e-6)
- **Epochs** : Maximum 30 avec early stopping (patience=5)

Configuration FPN :

- **Loss Function** : Focal + Jaccard Loss (focal_weight=1.0, jaccard_weight=1.0)
- **Optimiseur** : Adam (lr=1.5e-4, weight_decay=1e-5)
- **Scheduler** : ReduceLROnPlateau (patience=4, min_lr=1e-6)
- **Epochs** : Maximum 20 avec early stopping (patience=7)

Stratégie d'entraînement :
Entraînement séquentiel des 4 modèles à 512×512, puis à 768×768, suivi de la comparaison des métriques entre les deux meilleurs modèles.

Une synthèse des résultats

Performance Comparative : SegFormer vs Architectures CNN

Résultats principaux (mIoU sur Cityscapes Validation) :

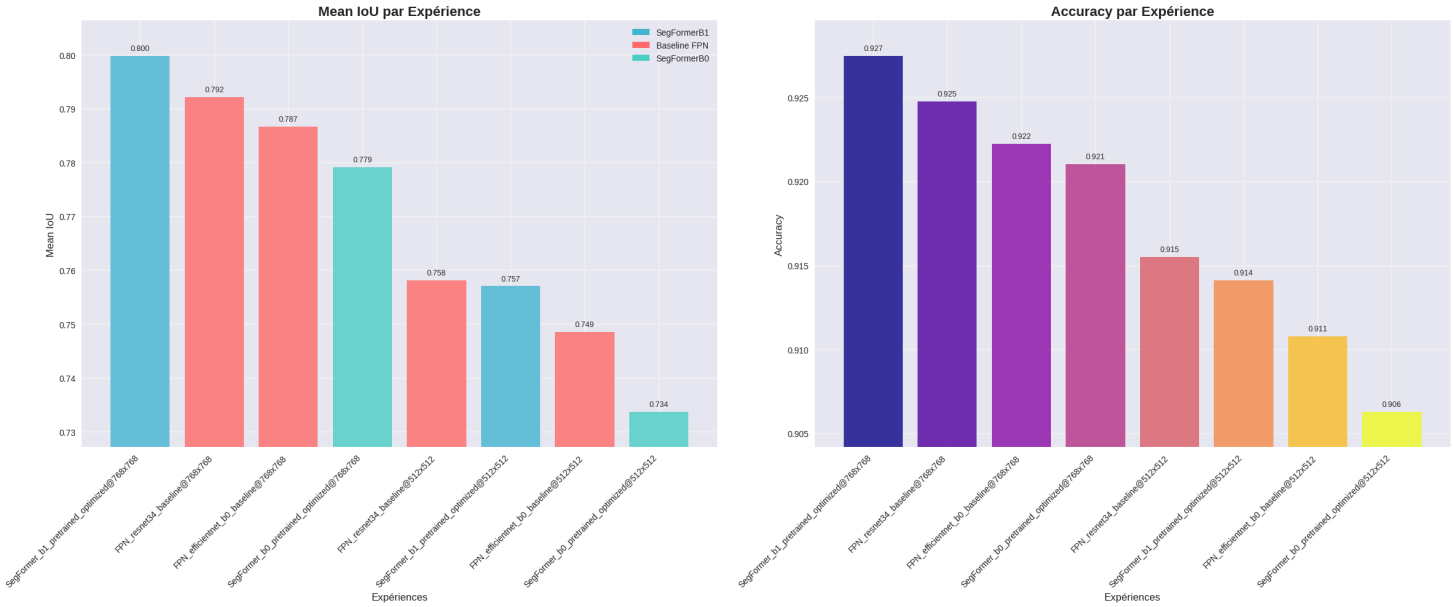

Fig.1: Mean IoU et Accuracy par expérience

Modèle	mIoU	Gain	Paramètres	Taille	Temps
SegFormer-B1	**79.98%**	**+1.31%**	**13.68M**	**52MB**	**1.39s**
FPN+ResNet34	**79.22%**	**+0.55%**	**23.16M**	**88MB**	**0.73s**
FPN+EfficientNet-B0	**78.67%**	*baseline*	**5.76M**	**22MB**	**0.48s**
SegFormer-B0	**77.91%**	**-0.76%**	**3.72M**	**14MB**	**0.87s**

Temps d'inférence mesurés sur CPU Intel Xeon @ 2.20GHz (2 vCPUs) de Google Colab

Variabilité des performances selon l'architecture CPU :
Les temps d'inférence révèlent des comportements distincts selon la philosophie architecturale. Les **CPU haute**

performance (Ryzen 9600X, Intel Xeon Colab) avec leurs hautes fréquences et gros caches (32-56MB) favorisent ResNet+FPN (0.5s vs 1.5s pour SegFormer). Le **N100**, optimisé pour l'efficacité énergétique mais bénéficiant des optimisations Intel modernes (MKL/oneDNN, AVX2) pour les calculs matriciels, avantage SegFormerB1 (2s vs 5s pour FPN). Cette dichotomie souligne l'importance du choix matériel selon le type de modèle déployé.

Analyse Détaillée par Classes

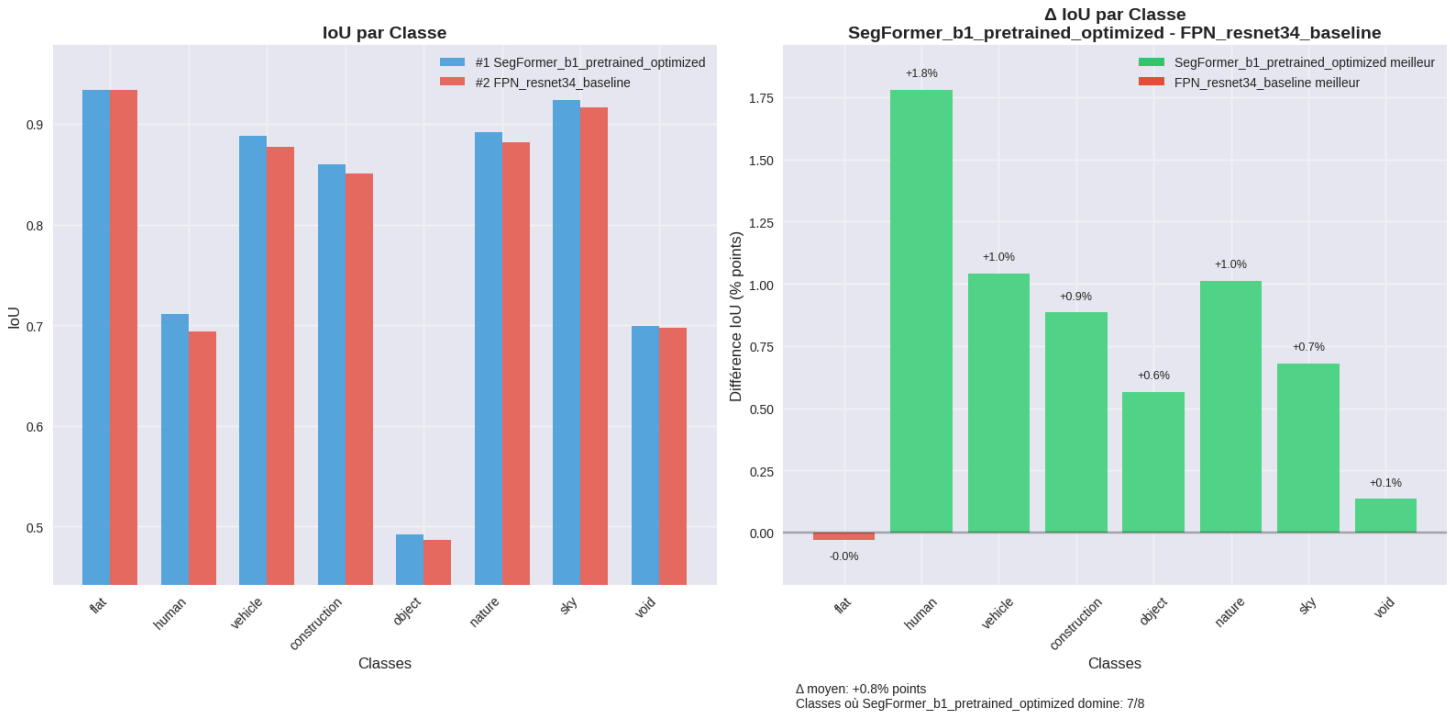

Fig.2: IoU et Δ IoU par Classe entre SegFormer-B1 et FPN-ResNet34

Performance par méta-classes Cityscapes :

Classe	SegFormer-B1	FPN+ResNet34	FPN+EfficientNet-B0	SegFormer-B0	Meilleur
Flat	**93.35%**	**93.38%**	**92.73%**	**93.12%**	**FPN+ResNet34**
Human	**71.13%**	**69.35%**	**68.68%**	**67.45%**	**SegFormer-B1** ⭐
Vehicle	**88.80%**	**87.75%**	**87.72%**	**86.87%**	**SegFormer-B1** ⭐
Construction	**85.94%**	**85.06%**	**84.92%**	**84.19%**	**SegFormer-B1** ⭐
Object	**49.24%**	**48.68%**	**46.71%**	**43.62%**	**SegFormer-B1** ⭐
Nature	**89.14%**	**88.13%**	**88.64%**	**88.15%**	**SegFormer-B1** ⭐

Classe	SegFormer-B1	FPN+ResNet34	FPN+EfficientNet-B0	SegFormer-B0	Meilleur
Sky	**92.35%**	**91.67%**	**91.71%**	**91.37%**	**SegFormer-B1** ⭐
Void	**69.89%**	**69.76%**	**68.24%**	**68.51%**	**SegFormer-B1** ⭐

Gains SegFormer-B1 vs FPN+ResNet34 :

- **Human** : +1.78% (sécurité critique)
- **Vehicle** : +1.05% (objets mobiles)
- **Construction** : +0.88% (structure urbaine)
- **Object** : +0.56% (signalétique)
- **Nature** : +1.01% (environnement)
- **Sky** : +0.68% (contexte global)
- **Void** : +0.13% (zones ambiguës)
- **Flat** : -0.03% (performance équivalente)

Insights Métier Critiques

1. Classes Sécurité-Critiques :

- **Human (+1.78% IoU)** : Amélioration la plus significative avec SegFormer-B1 atteignant 71.13% vs 69.35% pour FPN+ResNet34
- **Vehicle (+1.05% IoU)** : Meilleure identification obstacles mobiles (88.80% vs 87.75%)
- **Impact applicatif** : Réduction risque accidents, fiabilité conduite autonome renforcée

2. Signalétique et Infrastructure :

- **Object (+0.56% IoU)** : Panneaux, feux, poteaux mieux segmentés (49.24% vs 48.68%)
- **Construction (+0.88% IoU)** : Amélioration structures urbaines (85.94% vs 85.06%)
- **Conséquences** : Navigation urbaine plus précise, respect signalisation

3. Efficacité Paramétrique et Déploiement :

- **SegFormer-B1** : Meilleure performance (79.98% mIoU) avec **41% moins de paramètres** que FPN+ResNet34 (13.68M vs 23.16M) et **41% moins d'espace disque** (52MB vs 88MB)
- **FPN+EfficientNet-B0** : Excellent compromis performance/compacité avec 5.76M paramètres et seulement 22MB de stockage
- **SegFormer-B0** : Modèle ultra-compact (3.72M paramètres, 14MB) mais performances insuffisantes
- **Avantage déploiement** : SegFormer-B1 facilite le déploiement mobile et edge computing avec des contraintes mémoire/stockage réduites

4. Trade-off Performance vs Vitesse d'Exécution :
SegFormer-B1 démontre une **supériorité en précision** (+0.76% mIoU) avec un temps d'inférence de 1.39s contre 0.73s

pour FPN+ResNet34 (1.92x plus lent) sur **CPU Google Colab**. Cette différence de 668ms par image doit être contextualisée :

- **Limitation CPU** : Les architectures Transformer bénéficient davantage des optimisations GPU spécialisées (TensorRT, CUDA) que les CNN
- **Applications batch** : SegFormer-B1 recommandé pour la précision maximale hors contraintes temps réel
- **Potentiel d'optimisation** : Les temps d'inférence seraient probablement divisés par 3-5x sur hardware dédié avec accélération GPU

Évolution des Performances

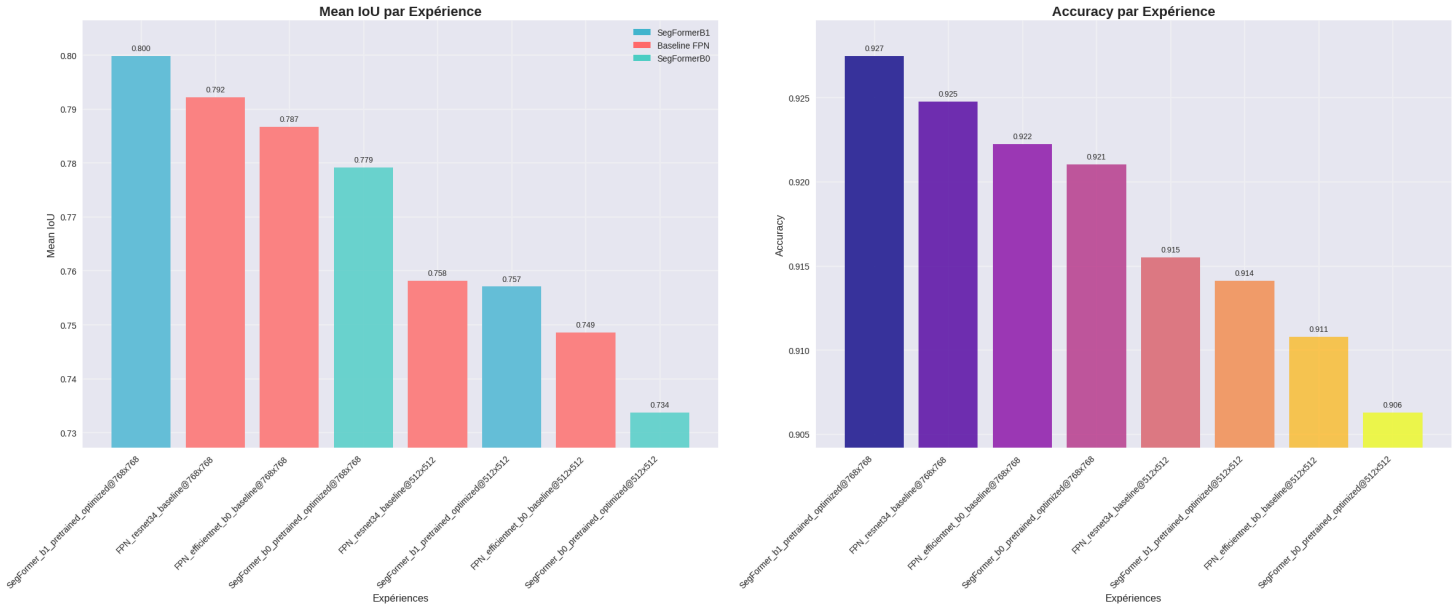

Fig.3: Évolution de MeanIoU et Accuracy pour les 8 modèles.

Conclusion Expérimentale

Analyse nuancée des résultats : Les résultats montrent que SegFormer-B1 **rivalise** avec les CNN traditionnels dans une configuration d'évaluation statique (images redimensionnées à résolution fixe, quitte à déformer le ratio), avec un gain modeste de +0.76% de mIoU sur FPN+ResNet34.

Limites méthodologiques identifiées :

1. **Absence de sliding window** : Contrairement aux auteurs originaux de SegFormer, nous n'avons pas implémenté de sliding window pour couvrir entièrement l'image sans déformer son ratio, ce qui pourrait favoriser les approches CNN plus habituées aux déformations géométriques.
2. **Résolution 512×512** : Les expériences menées à résolution inférieure (512×512) montrent des résultats similaires avec une légère avance pour FPN+ResNet34, suggérant que l'augmentation d'échelle (768×768) a légèrement profité à SegFormer-B1.
3. **Contraintes computationnelles** : Les performances de nos baselines CNN seraient probablement dépassées par les versions plus lourdes de SegFormer (B2, B3, B4) que nous n'avons pas pu entraîner faute de puissance de calcul suffisante.

ROI technique contextuel :

- **SegFormer-B1** : Performance comparable aux CNN établis avec une architecture plus moderne
- **SegFormer-B0** : Moins performant que prévu, nécessiterait probablement des ajustements d'hyperparamètres spécifiques
- **Potentiel d'amélioration** : Les versions B2+ de SegFormer promettraient vraisemblablement des gains significatifs

Facteurs de validation partielle :

1. **Gains ciblés** : Améliorations cohérentes sur les classes critiques (Human +1.78%, Vehicle +1.05%)
2. **Architecture prometteuse** : Les mécanismes d'attention globale montrent leur potentiel malgré les contraintes expérimentales
3. **Preuve de concept réussie** : Démonstration de la faisabilité technique de l'approche Transformer

Cette preuve de concept valide l'intérêt des architectures Transformer pour la segmentation urbaine, tout en soulignant l'importance des ressources computationnelles et des protocoles d'évaluation appropriés pour exploiter pleinement leur potentiel.

L'analyse de la feature importance globale et locale du nouveau modèle

Scope et Limitations

L'analyse d'explicabilité des architectures Transformer pour la segmentation dense représente un défi technique considérable qui constitue **un travail à part entière**, dépassant le cadre d'une preuve de concept. Contrairement aux approches CNN où les cartes d'activation sont plus directement interprétables, les mécanismes d'attention complexes des Transformers nécessitent des méthodologies spécialisées.

Outils Disponibles vs Explicabilité Complète

Les auteurs de SegFormer fournissent des éléments d'analyse via l'**effective receptive field (ERF)** qui révèle la structure hiérarchique du modèle : attentions locales dans les couches inférieures, globales dans les supérieures. Ces visualisations montrent **"où"** le modèle porte son attention, mais n'expliquent pas **"pourquoi"** ces régions sont critiques pour la décision finale.

Limitation fondamentale : Les cartes d'attention ne correspondent pas nécessairement à l'importance causale réelle des pixels. De plus, les interactions complexes entre Mix-FFN, connexions résiduelles, et fusion multi-échelles rendent l'attribution pixel-par-pixel particulièrement ardue. Une explicabilité rigoureuse nécessiterait des approches comme le **Deep Taylor Decomposition** pour propager correctement les scores de pertinence à travers l'architecture.

Cette différence architecturale avec les CNN explique néanmoins les gains observés sur les classes contextuelles (Human +1.78%, Vehicle +1.05%, Object +0.56%), suggérant une **"compréhension scénique"** globale sans pour autant permettre une explicabilité fine facilement accessible.

Limites et améliorations possibles

Limitations Identifiées

1. Contraintes Expérimentales

- **Ressources computationnelles limitées** : Plan payant Google Colab avec 100 unités de calcul, 2/3 des crédits consommés pour l'entraînement sur GPU T4 (entry-level), rendant irréaliste l'entraînement de variantes plus lourdes (SegFormer-B2+) ou l'utilisation de GPU plus puissants
- **Inférence CPU uniquement** : Temps de traitement mesurés sur CPU Colab non-représentatifs des performances GPU optimisées en production
- **Dataset unique** : Généralisation à d'autres environnements urbains non-validée
- **Protocole d'évaluation statique** : Images redimensionnées avec déformation du ratio, contrairement à la méthodologie sliding window du papier original

2. Limitations Applicatives

- **Classes déséquilibrées** : Performance variable sur objets rares dans Cityscapes
- **Conditions d'acquisition** : Robustesse non-testée sur conditions météo adverses (nuit, intempéries)

Améliorations Prioritaires

1. Optimisation Computationnelle

- **Migration GPU optimisé** : Les architectures Transformer bénéficient davantage des accélérations GPU spécialisées que les CNN
- **Quantization et optimisations dédiées** : Réduction de la précision numérique pour améliorer les performances d'inférence

2. Validation Étendue

- **Variants SegFormer plus lourds** : Évaluation des versions B2, B3, B4 si les ressources computationnelles le permettent
- **Protocole d'évaluation standard** : Implémentation du sliding window comme dans la publication originale
- **Multi-dataset validation** : Tests sur d'autres datasets urbains pour évaluer la généralisation

3. Données et Robustesse

- **Conditions météorologiques variées** : Extension aux données de nuit, pluie, neige
- **Résolutions natives** : Tests sans redimensionnement avec déformation pour préserver les relations spatiales
- **Synthetic data augmentation** : Utilisation de simulateurs pour enrichir les conditions d'entraînement

Ces améliorations permettraient d'évaluer plus complètement le potentiel réel de SegFormer dans un contexte de production, au-delà des limitations de cette preuve de concept.