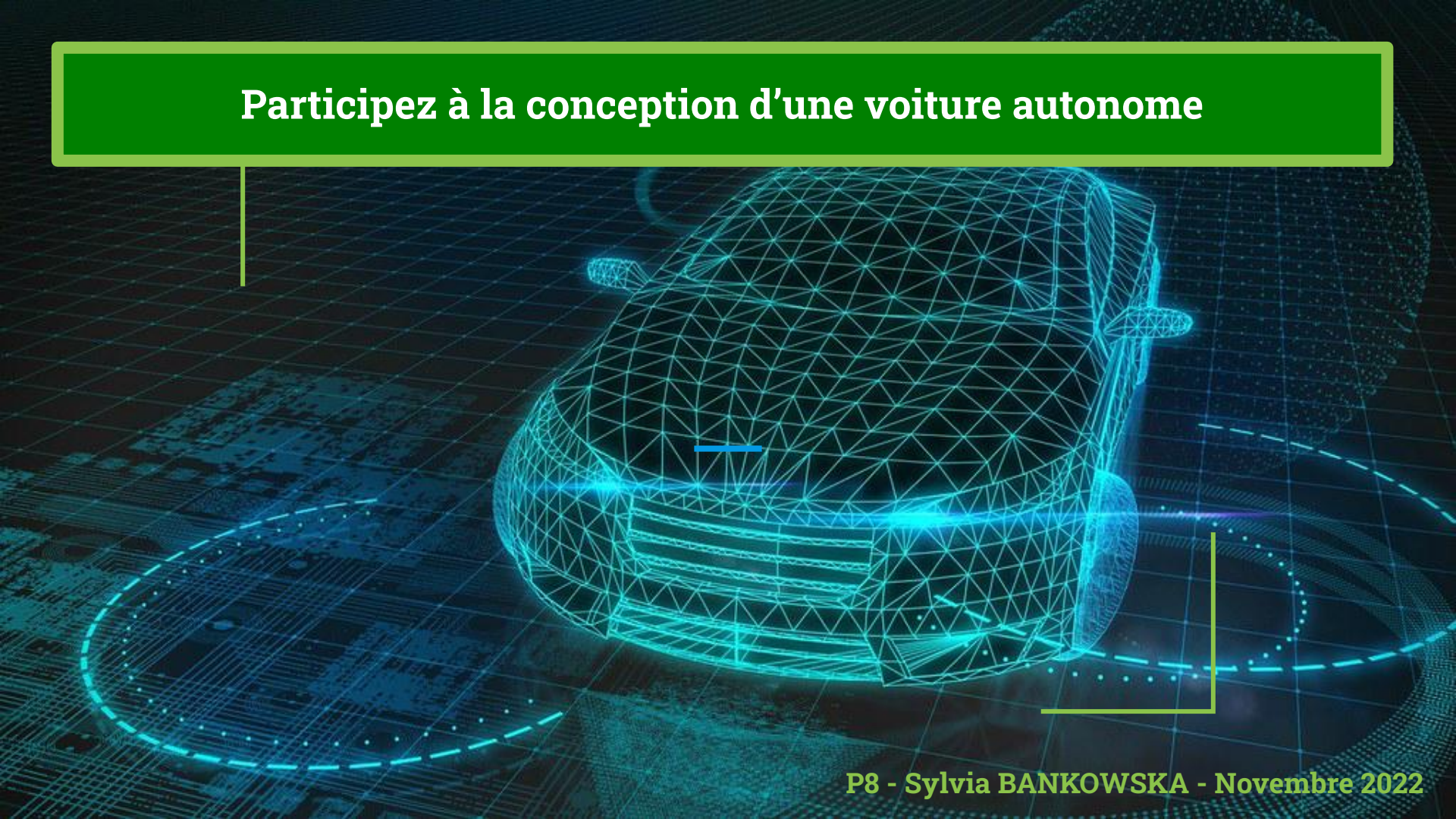


Participez à la conception d'une voiture autonome



Sommaire

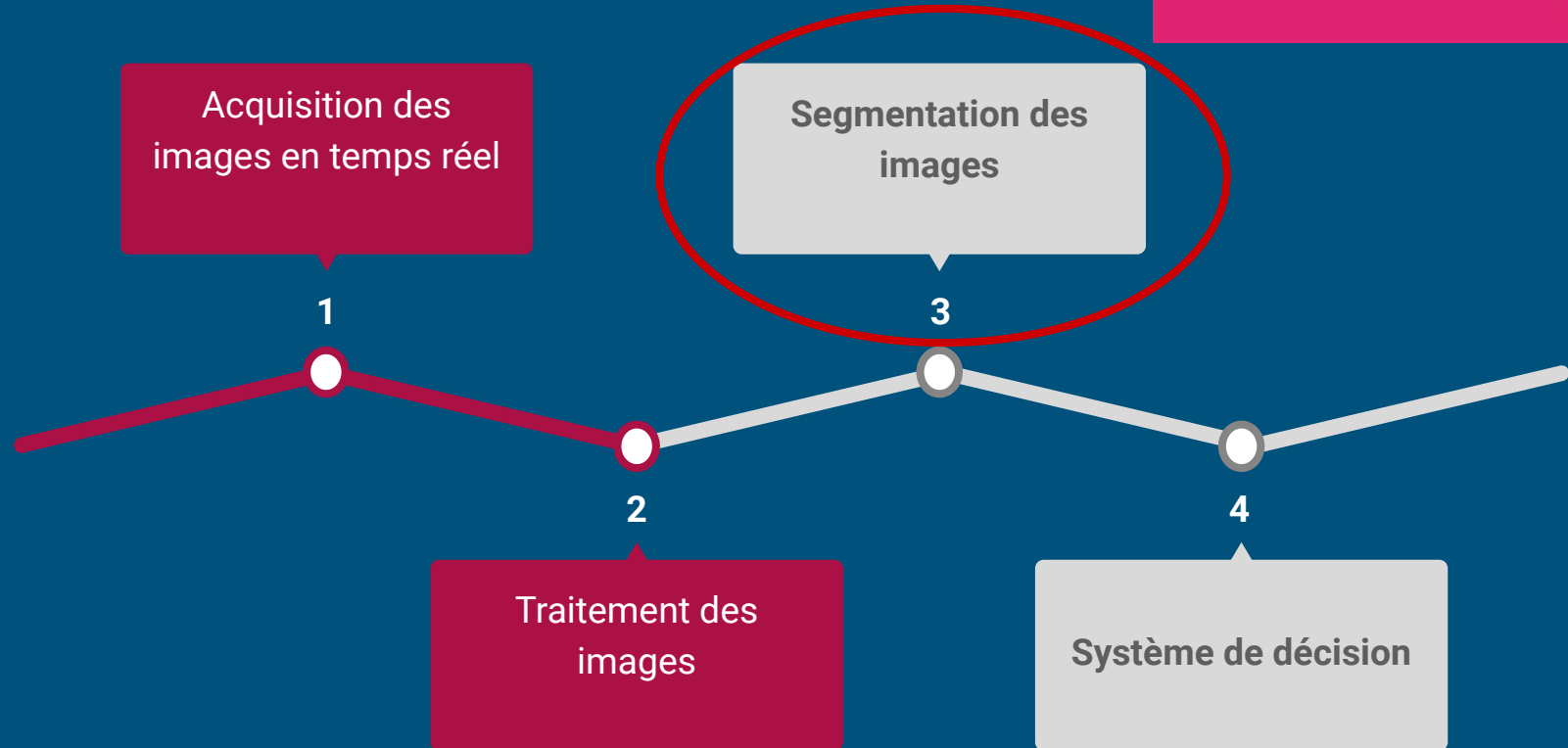


1. Enjeux & Objectifs
 2. Jeu de données
 3. Flux de données
 4. Métriques & Loss functions
 5. Baseline
 6. Modélisations
 7. Synthèse
 8. Modèle final
 9. Déploiement
 10. Conclusions
-

Enjeux & Objectifs



Future Vision Transport



Objectifs

Modèle de segmentation sémantique



Transfer Learning :

- Backbone
- Poids par Imagenet

Image



Mask



Prédiction



Jeu de données

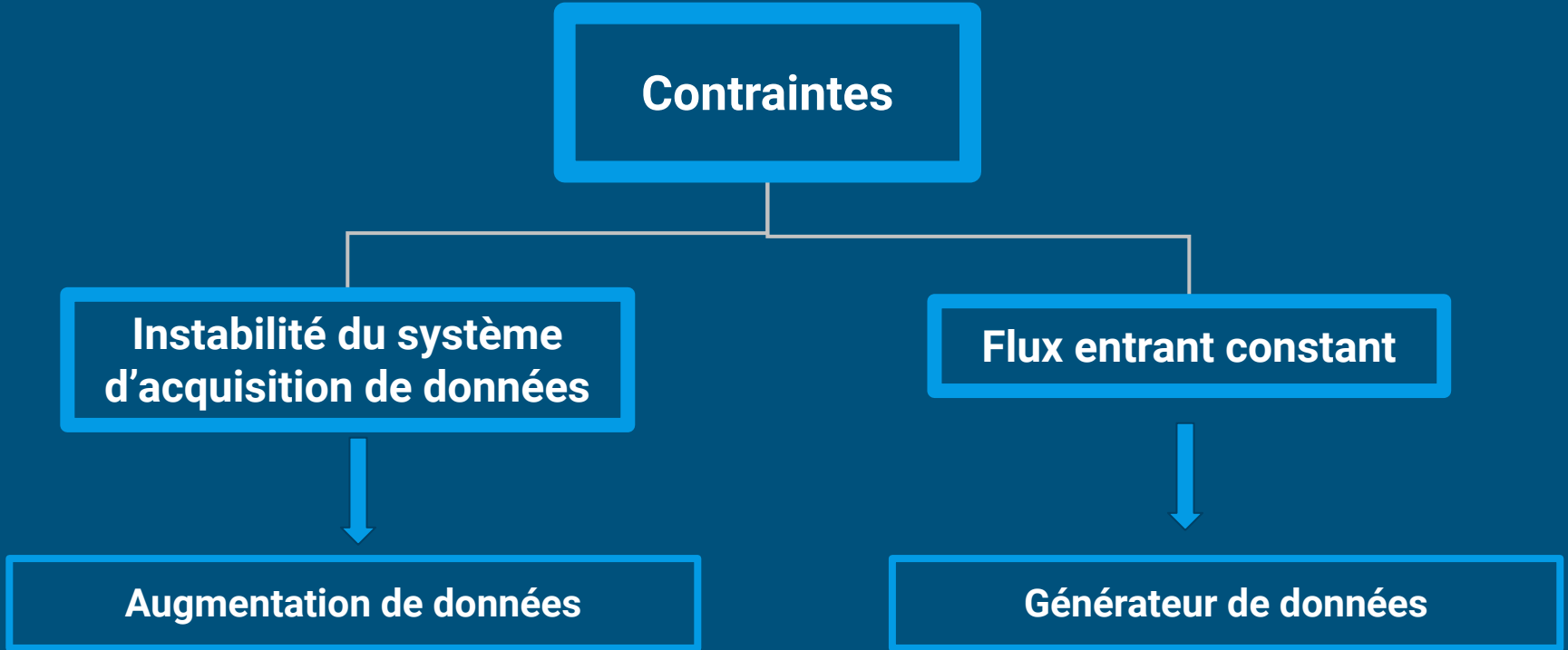
01	Train set	<ul style="list-style-type: none">• 2975 items x 2• Images & Masks
02	Validation set	<ul style="list-style-type: none">• 500 items x 2• Images & Masks
03	Test set	<ul style="list-style-type: none">• 1525 items x 2• Images & Masks

CITYSCAPES :

- Caméras embarquées dans 50 villes allemandes
- Conditions météorologiques variées
- 25 000 photos : routes, véhicules, objets routiers, etc.
- 32 sous-catégories & 8 catégories principales



Flux de données



Augmentation de données

- Création de nouvelles images artificielles
- Éviter le risque de sur-apprentissage
- Pallier les défauts de prise d'image

Photo
originale



Blur



Zoom



Brightness



Générateur de données

Input

Montée rapide du volume des données

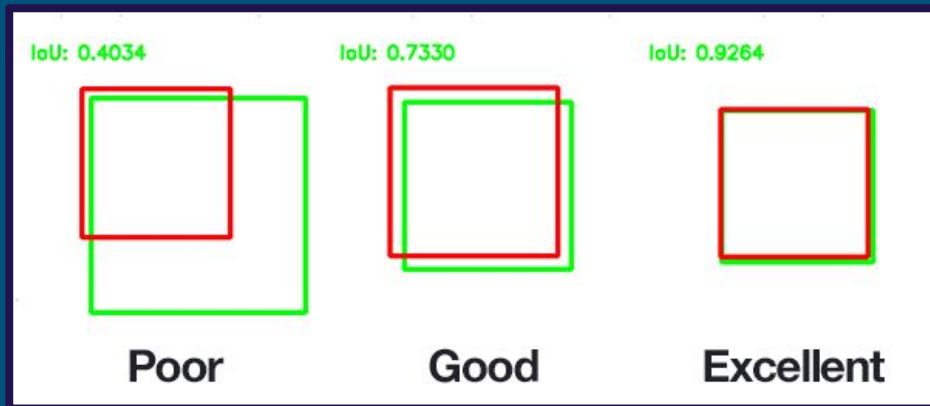
Load

Eviter de charger toutes les données

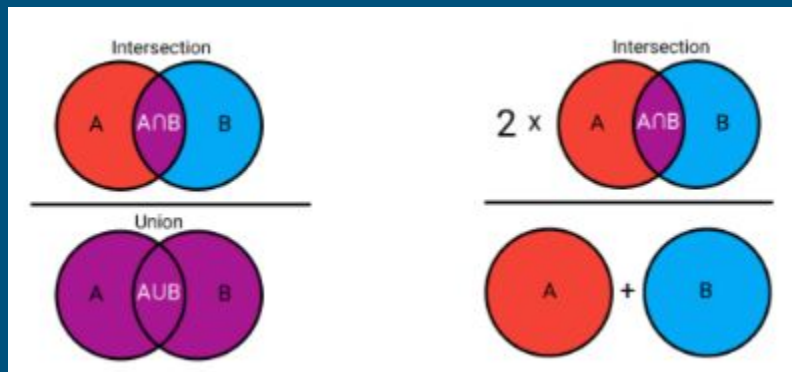
Data Generator

- ❑ Classe de type Sequence :
 - ❑ Chargement de données
 - ❑ Attribution de classes-cibles
 - ❑ Distribution des images par batches
 - ❑ Augmentation de données

Métriques



IoU (Jaccard) vs Dice (F1)



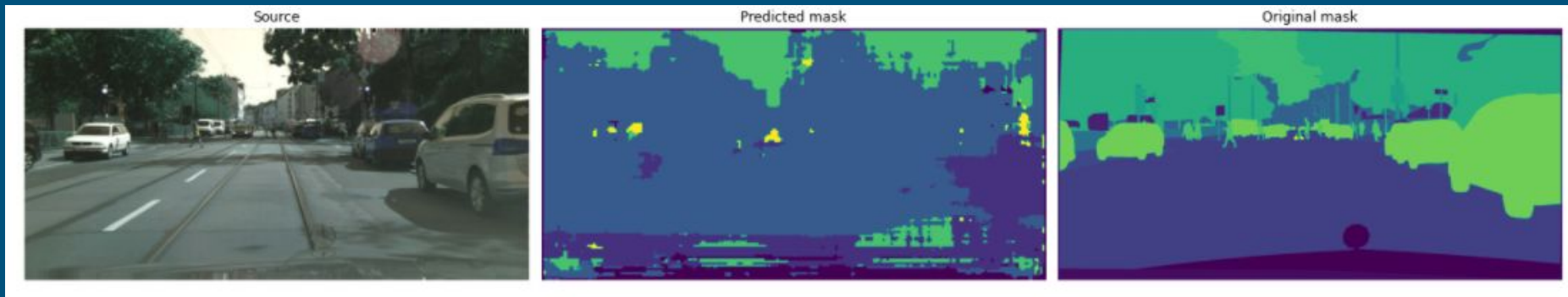
Ratio entre intersection
des masques réels et prédits
et union des 2 masques.

Loss functions

- **Dice Loss**
- **Weighted Dice**
- **Focal Loss**
- **BCE Jaccard**
- **Categorical Focal + Dice Loss**

Model	IoU	Dice	training_time
Base-Categ-Focal-Dice	0.496615	0.601135	7261.540968
Base-Dice	0.513233	0.622157	7420.738188
Base-Weight-Dice	0.387483	0.507299	7506.385689
Base-Focal	0.423833	0.539918	7538.611242
Base-Jaccard	0.449309	0.561707	7876.374853

Baseline



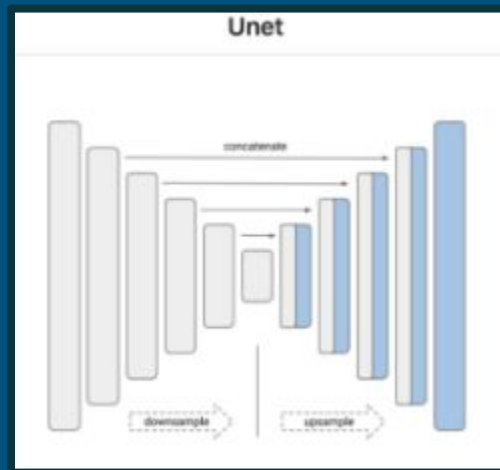
U-Net sans augmentation de données :

- IoU : 0.496
- F1 Score : 0.6
- Loss : 0.466 (Categorical Focal + Dice Loss)

Modélisations

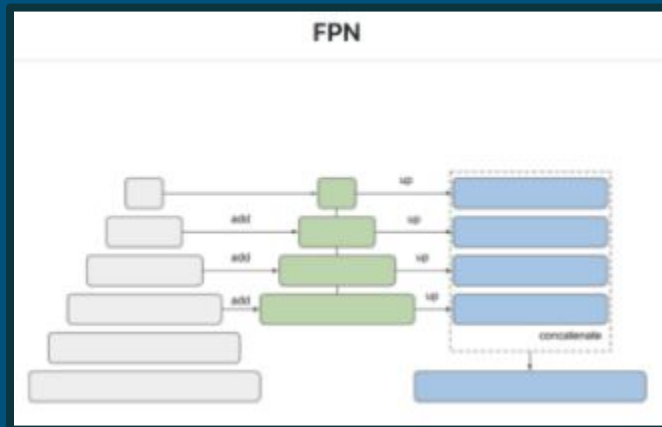
U-Net + ResNet152

- avec augmentation
- sans augmentation



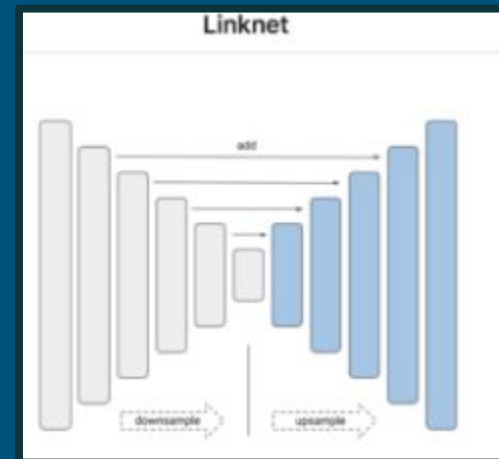
FPN + ResNet152

- avec augmentation
- sans augmentation



Linknet + Resnet152

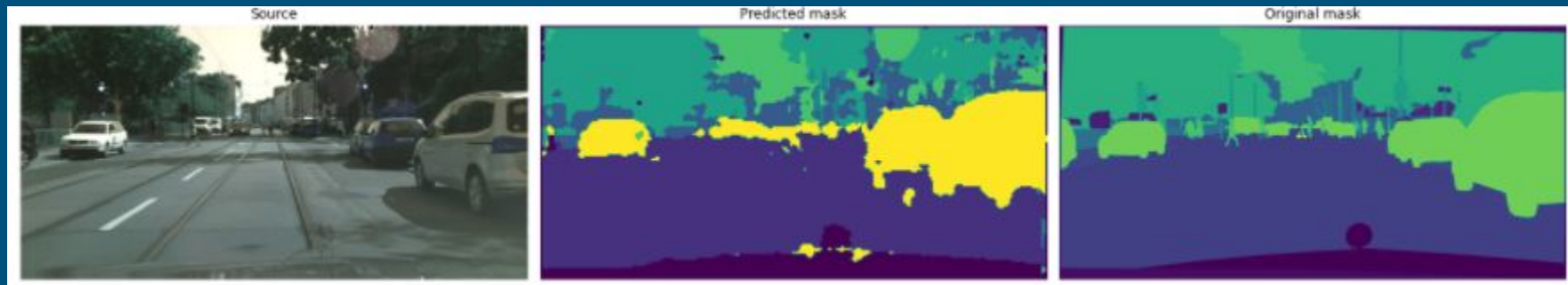
- avec augmentation
- sans augmentation



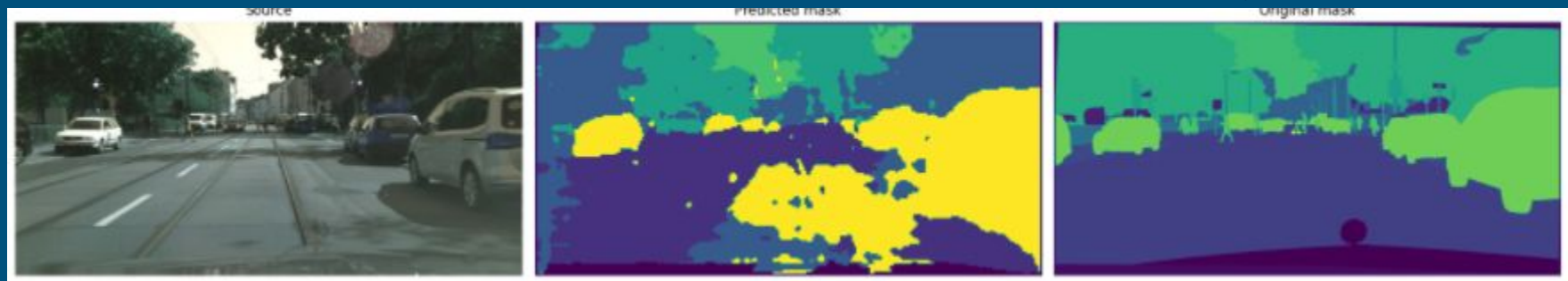
Synthèse

Model	IoU	Dice	training_time
Base-Categ-Focal-Dice	0.496615	0.601135	7261.540968
Base-Dice	0.513233	0.622157	7420.738188
Base-Weight-Dice	0.387483	0.507299	7506.385689
Base-Focal	0.423833	0.539918	7538.611242
Base-Jaccard	0.449309	0.561707	7876.374853
Unet-Resnet-NOaugm	0.611504	0.719118	10515.987629
Unet-Resnet-augm	0.635118	0.742667	21435.816270
FPN-Resnet-NOaugm	0.607629	0.721474	10060.494859
FPN-Resnet-augm	0.645746	0.753668	21951.161641
Linknet-Resnet-NOaugm	0.608704	0.721641	9976.627367
Linknet-Resnet-augm	0.585186	0.702199	20616.900008
Best_with_Dice	0.653487	0.761291	21772.060686

Modèle final



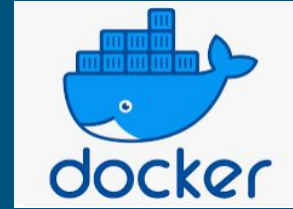
Categorical Focal + Dice Loss vs Dice Loss



Déploiement



Déploiement



Docker image



Install Python / pip3

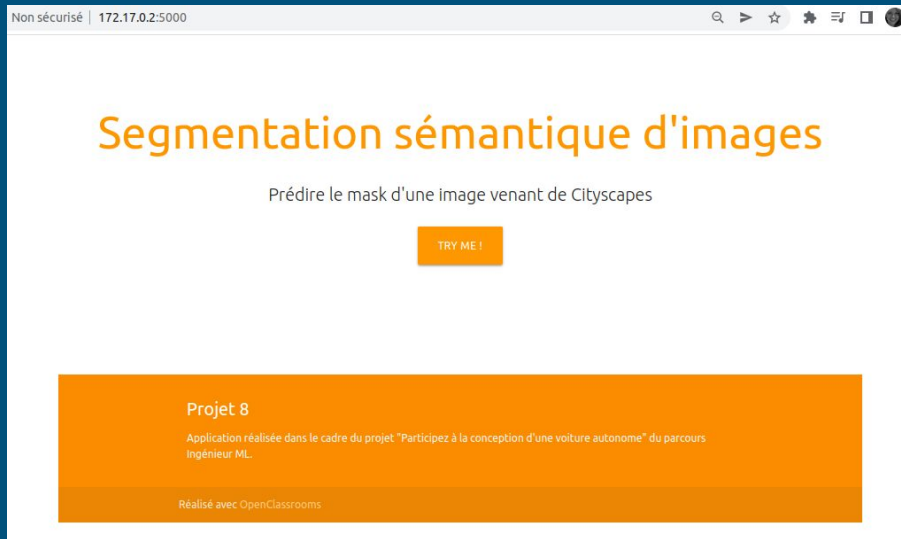
Copie code source

Install Modules

Expose Port

Container en mode
exécutable

Docker Container
<http://172.17.0.2:5000/>

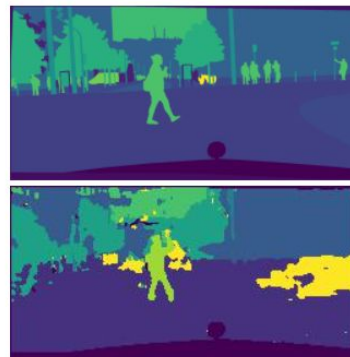


<http://sylviaban.pythonanywhere.com/>

ngrok

Résultat de la prédiction :

Mask original versus mask prédit



Conclusions

Les pistes d'amélioration

1. **Échantillon** plus volumineux
2. Utilisation d'autres **architectures**
3. Utilisation d'autres **backbones**
 - a. `#BACKBONE = 'vgg16'`
 - b. `#BACKBONE = 'vgg19'`
 - c. `#BACKBONE = 'densenet201'`
 - d. `#BACKBONE = 'densenet121'`
 - e. `#BACKBONE = 'efficientnetb0'`
 - f. `#BACKBONE = 'efficientnetb7'`
 - g. `#BACKBONE = 'inceptionv3'`
4. Tester d'autres **loss functions**
5. Tester d'autres méthodes **d'augmentation de données**
6. **Taille** max des images

FIN

Avez-vous des questions ?

