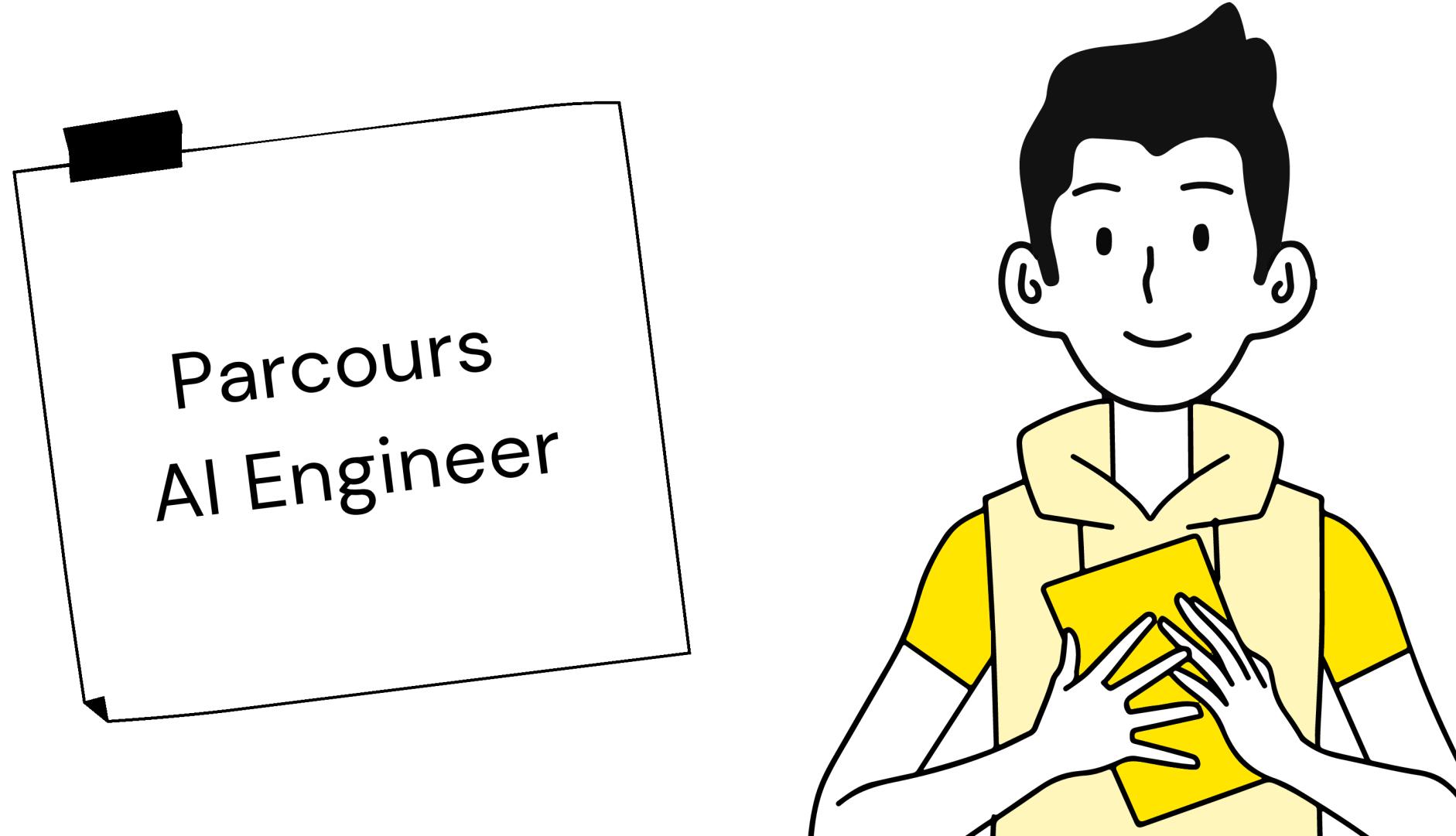
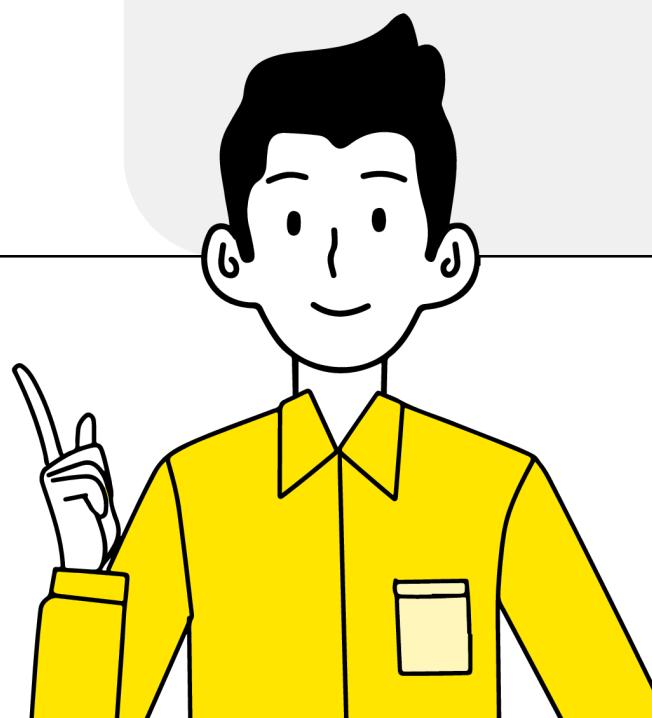


# Traitez les images pour le système embarqué d'une voiture autonome.



**Notre mission:**

**Concevoir un premier modèle  
de segmentation d'images**



# Les étapes d'analyse

## Partie 1

- Présentation du contexte
- Prise en main

## Partie 2

Mise en place des modèles

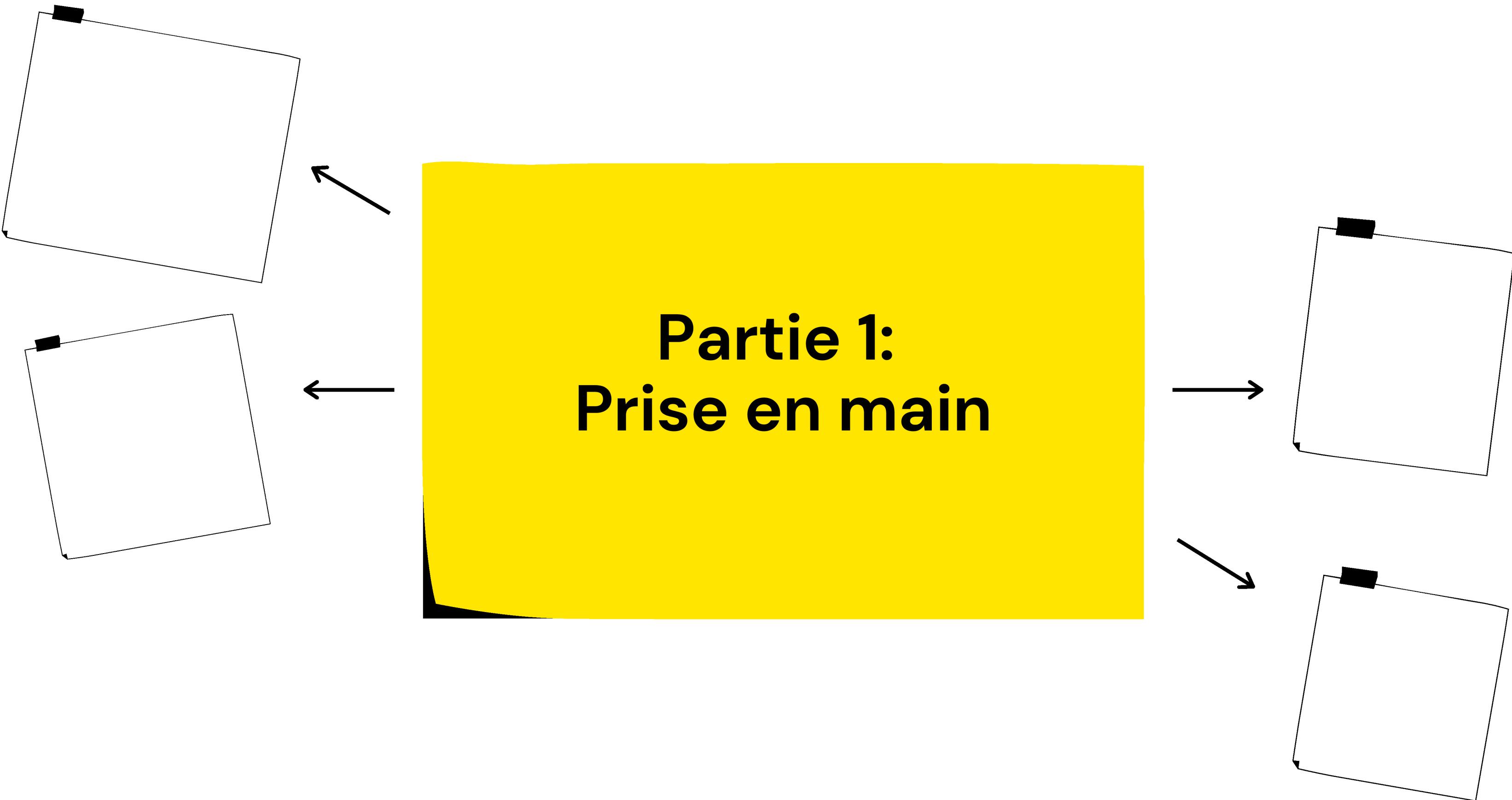
## Partie 3

Déploiement et Test

## Partie 4

Conclusion

# **Partie 1: Prise en main**



# Partie 1: La situation

**Qui sommes nous**

**Future Vision Transport**

**Notre rôle dans le  
projet**

**Responsable du modèle de segmentation**

**Etat de l'art**

**Notre exercice est une 1ère étape d'un  
concept beaucoup plus vaste.**



# Partie 1: Prise en main

## Ressources

2 dossiers de 3500 et 20 000 fichiers:

- Dans 3 dossiers (train, test, val)
- Dans des dossiers de villes
- Série de 3 images + un fichier JSON



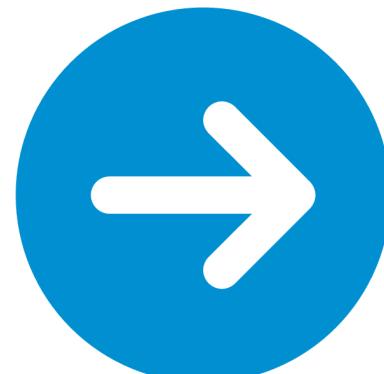
# Partie 1: Prise en main

Principe de segmentation

Observer les différents éléments d'une image



D'une photo...



...une décomposition de groupes



## **Partie 2: Mise en place des modèles**



# Partie 2: Mise en place des modèles

**Modèle 1**

**U-Net**

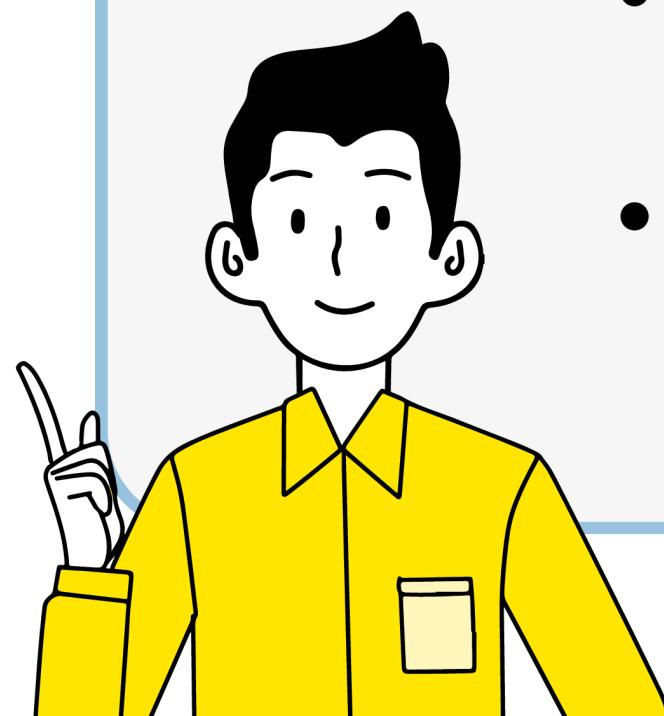
**Modèle 2**

**DeeplabV3**

## Préparation Commune

# De mise en place

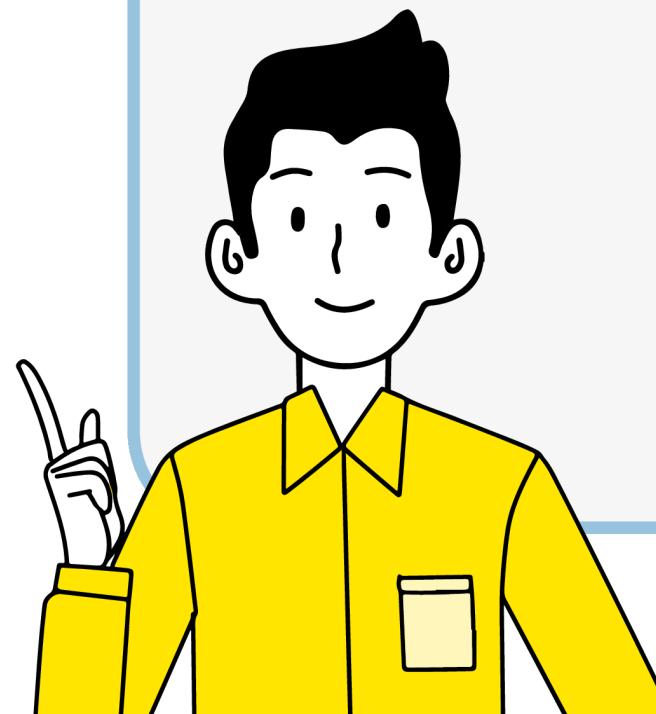
- **Créer une limitation dans le nombre de dossier utilisé.**
- **Création des masques avec les fichiers JSON et nos labels.**
  - On nous demande 8 labels différents
  - Route, Personne, Voiture, Bâtiment, Végétation, Ciel, Poteau, Sol
- **Lancement du traitement dans les dossiers. (Train, Test)**
- **Traitement des dossiers villes.**
- **Normalisation et One-Hot Encoding des mask.**
- **Utilisation de tf.data.AUTOTUNE pour exploiter tous les coeurs CPU.**
- **Utilisation de sparse\_categorical\_crossentropy**



## Préparation Commune

## De test

- Calcul de l'Intersection over Union (IoU).
- Calcul du coefficient de Dice.
- Graphique de l'évolution de l'IoU / Epoch.
- Visualisation des prédictions.

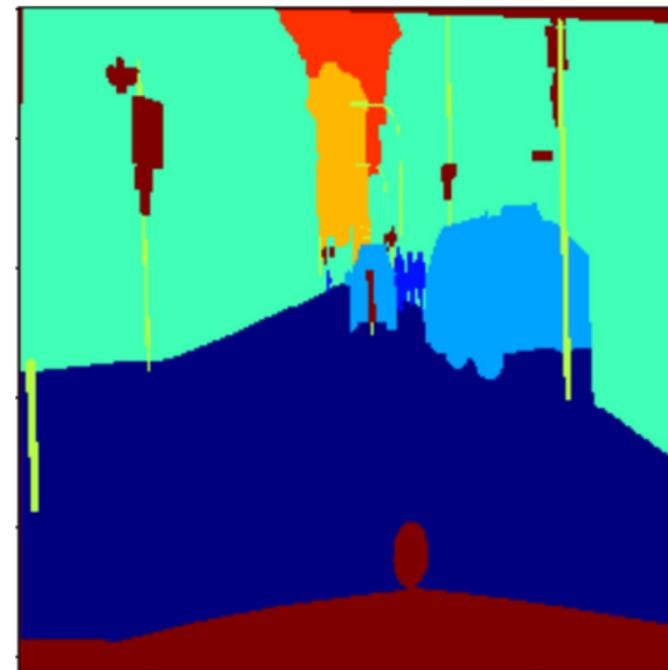


## Partie 2: Mise en place des modèles

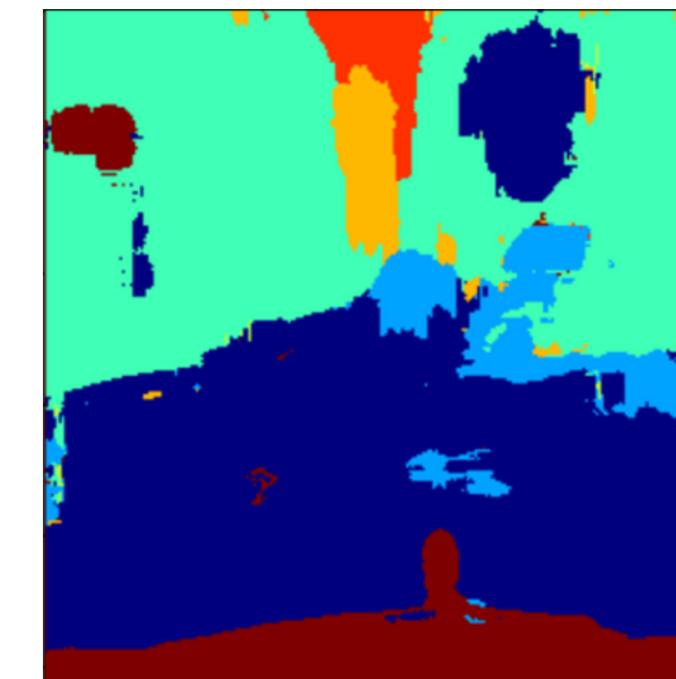
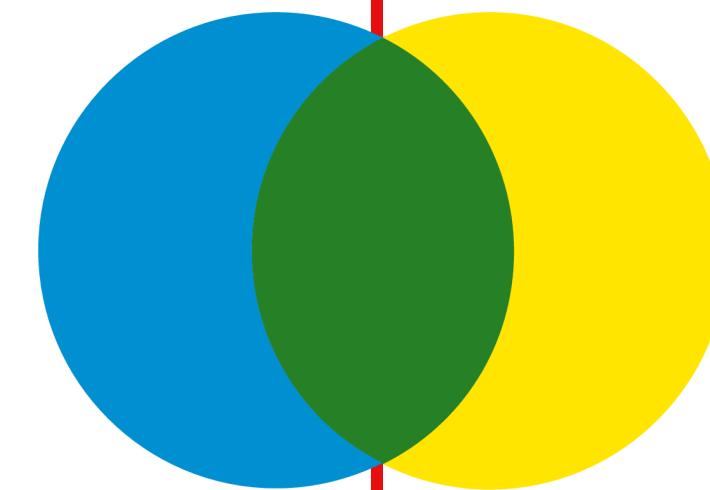
Préparation  
Commune

### Les nouveaux indicateurs : IoU et Dice Coefficient

Masque  
réel



Ensemble de pixel



Masque  
prédit

Intersection Over Union (IoU): Score entre 0 et 1. Calcul le ratio entre

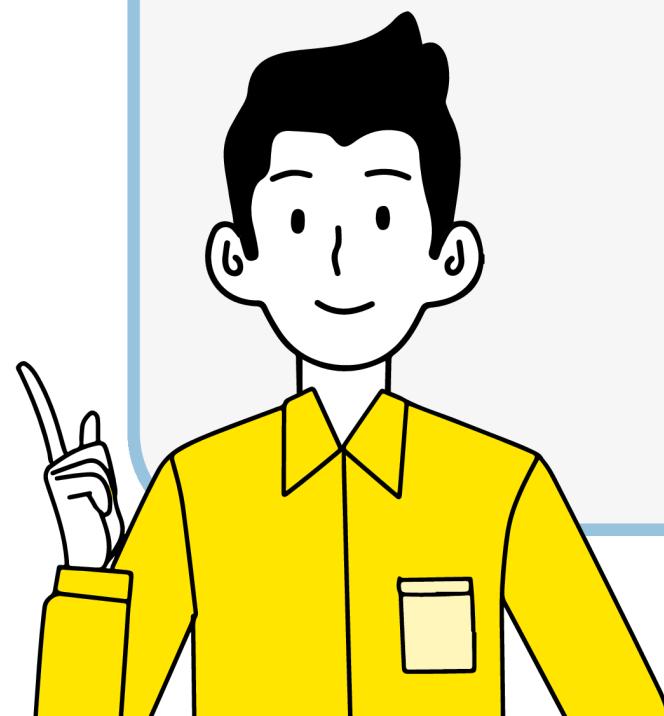


Dice Coefficient: Score entre 0 et 1. Mesure la zone de correspondance :



## Nos leviers d'améliorations

- Ajout d'une Data Augmentation.
- Taille des images.
- Learning rate.
- Paramètres classique. (Batch Size, Epoch,...)



⚠️ Augmentation très importante  
du temps d'entraînement.

## Partie 2: Mise en place des modèles

### Modèle 1

### U-Net

#### Force

- Architecture simple et efficace.
- Excellente performance pour la segmentation biomédicale.

#### Faiblesse

- Limité aux résolutions d'entrée fixes.
- Problèmes de surajustement.



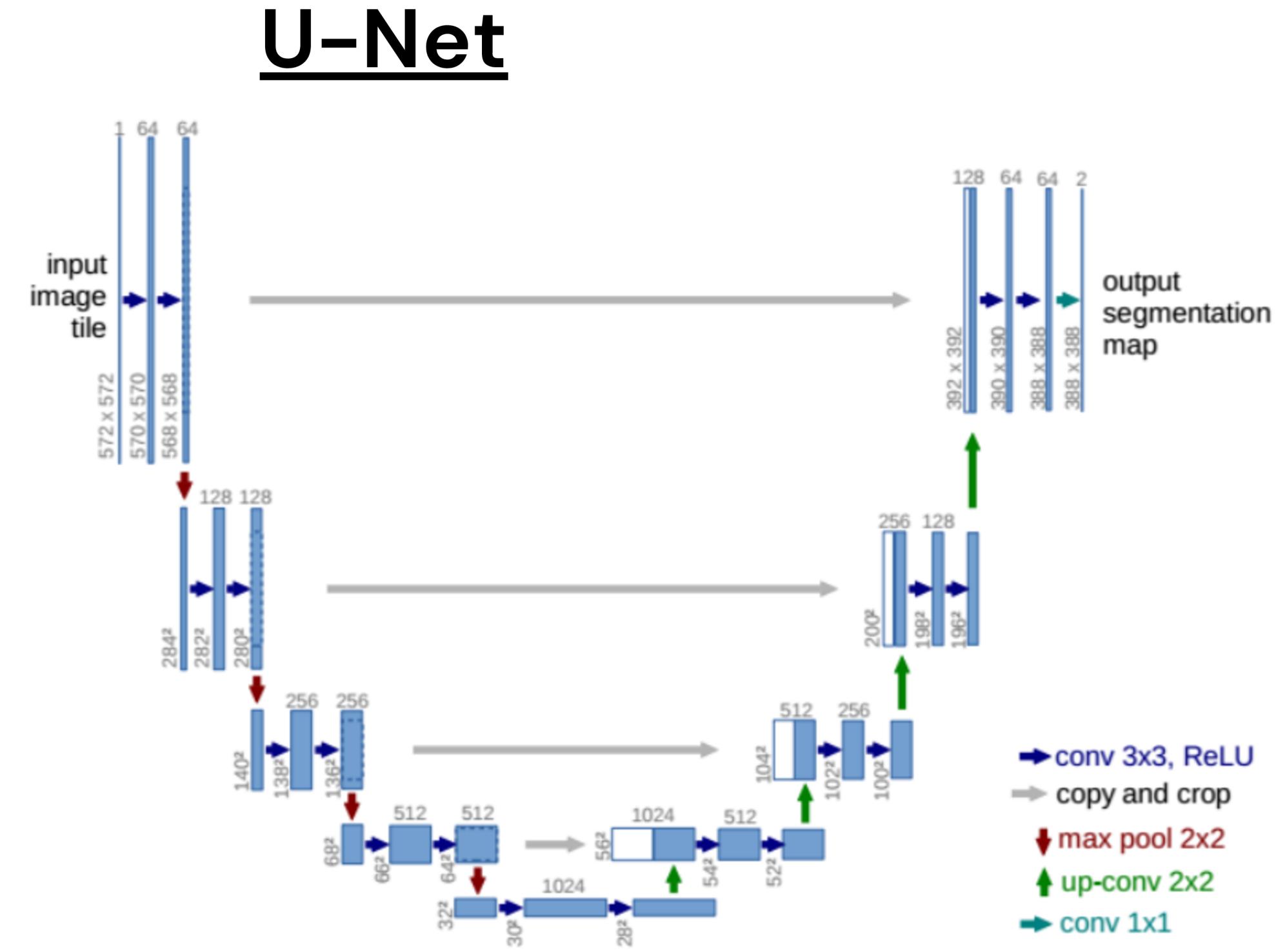
# Partie 2: Mise en place des modèles

## Modèle 1

Fonctionnement

Chemin d'encodage  
Gauche

Chemin décodage  
Droite

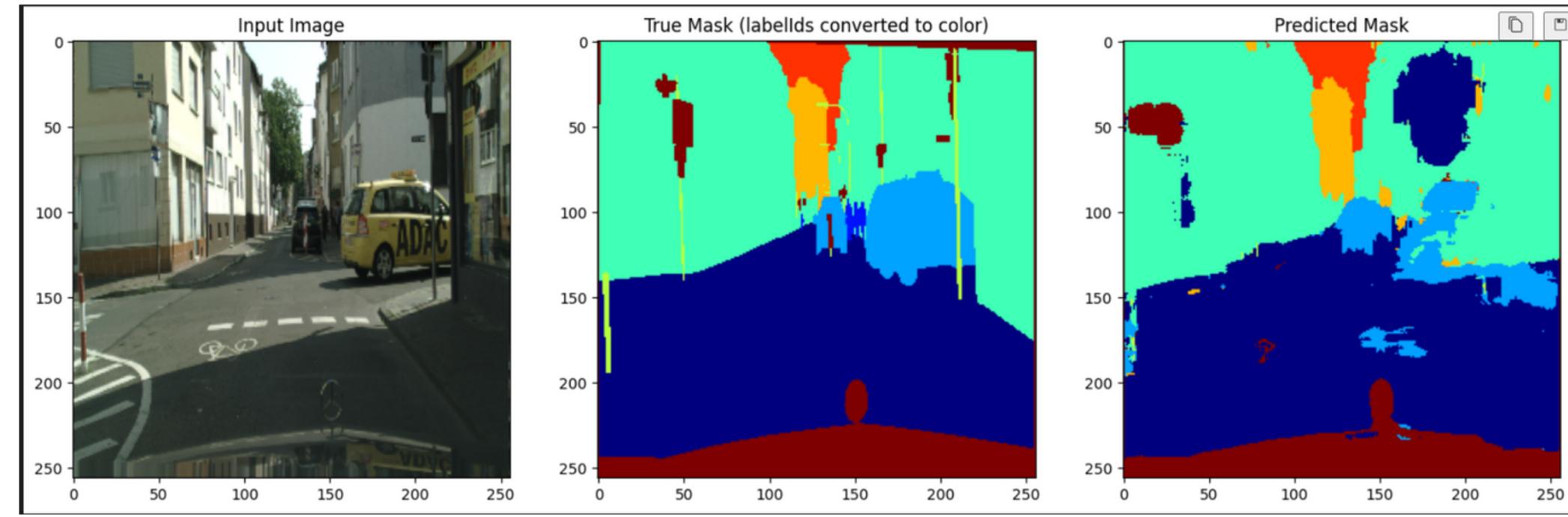


# Partie 2: Mise en place des modèles

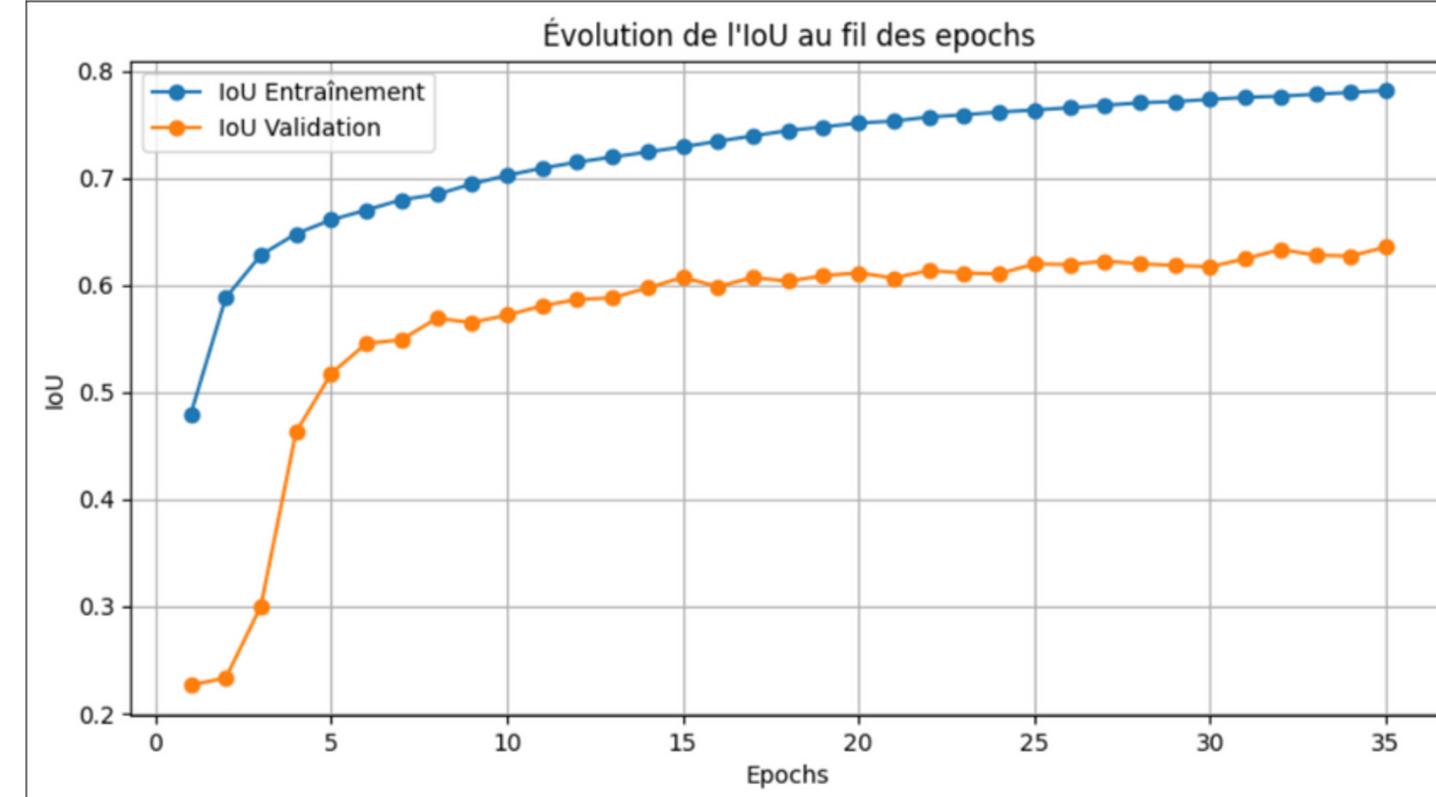
## Modèle 1

## U-Net

Simulation d'une prédiction



Evolution de la métrique IoU



## Partie 2: Mise en place des modèles

Modèle 2

DeeplabV3

Force

- Segmentation précise avec ASPP.
- Utilisation de convolutions dilatées.

Faiblesse

- Complexité de l'architecture.
- Grande consommation de mémoire.



# Partie 2: Mise en place des modèles

## Modèle 2

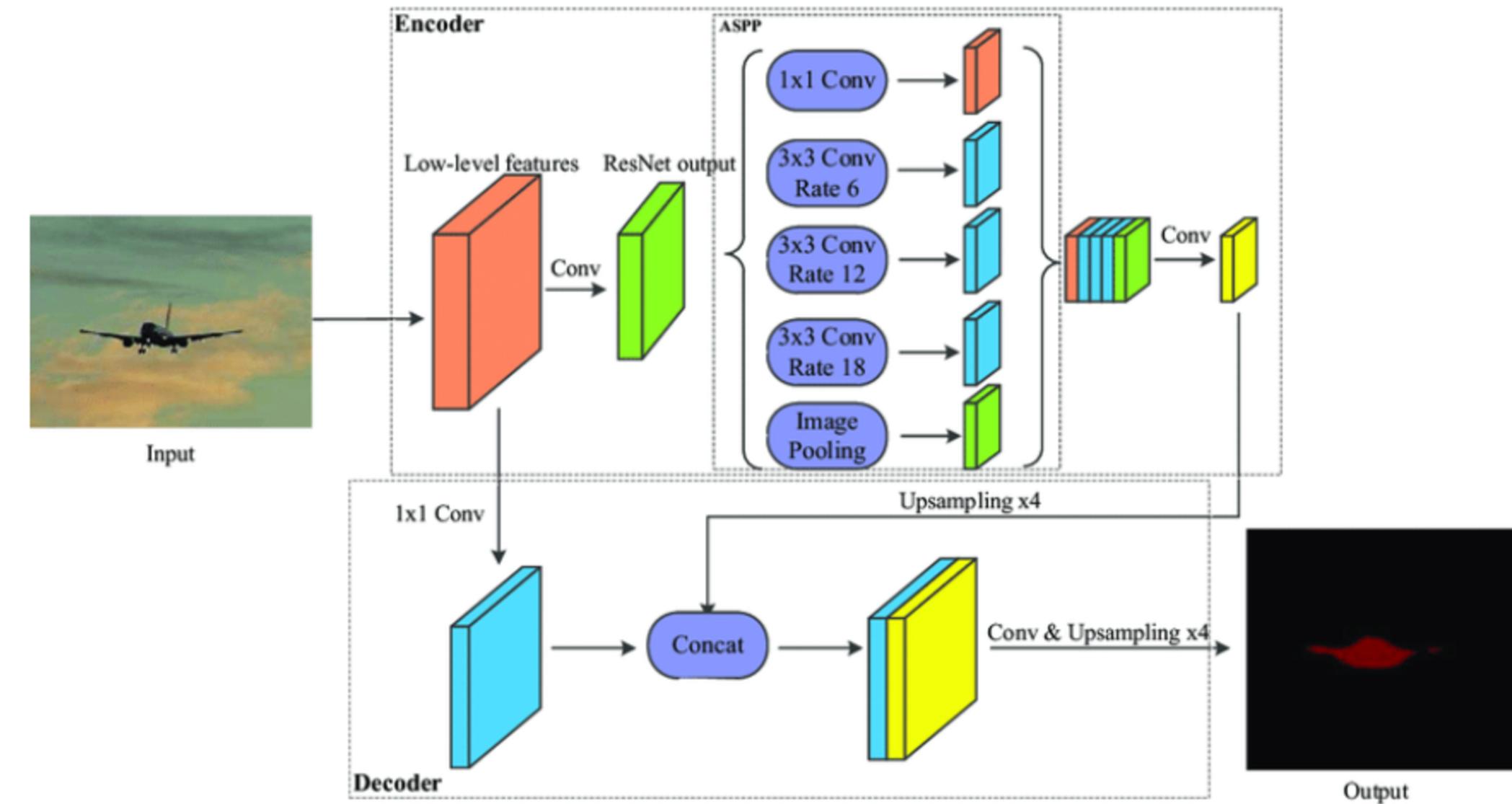
Fonctionnement

Encodage : ASPP

Décodage : Utilisation de l'ASPP

## DeeplabV3

Utilisation de ResNet



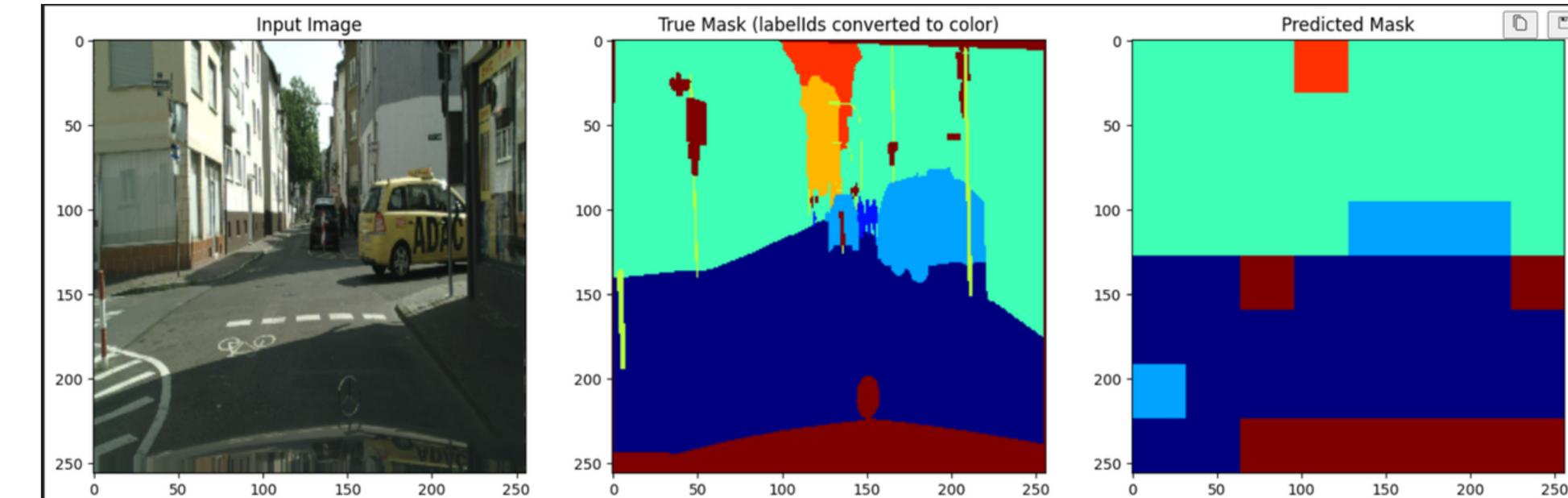
# Partie 2: Mise en place des modèles

Modèle 2

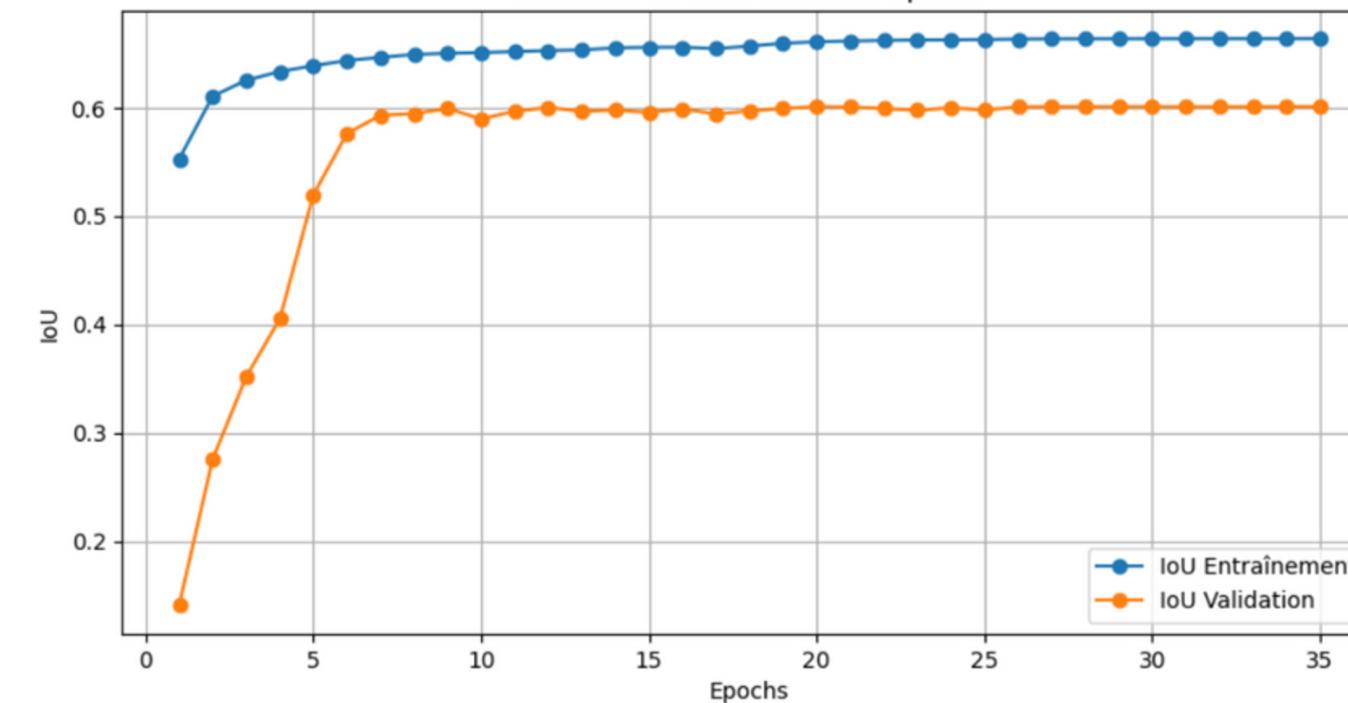
DeeplabV3

Utilisation de ResNet

Simulation d'une prédiction



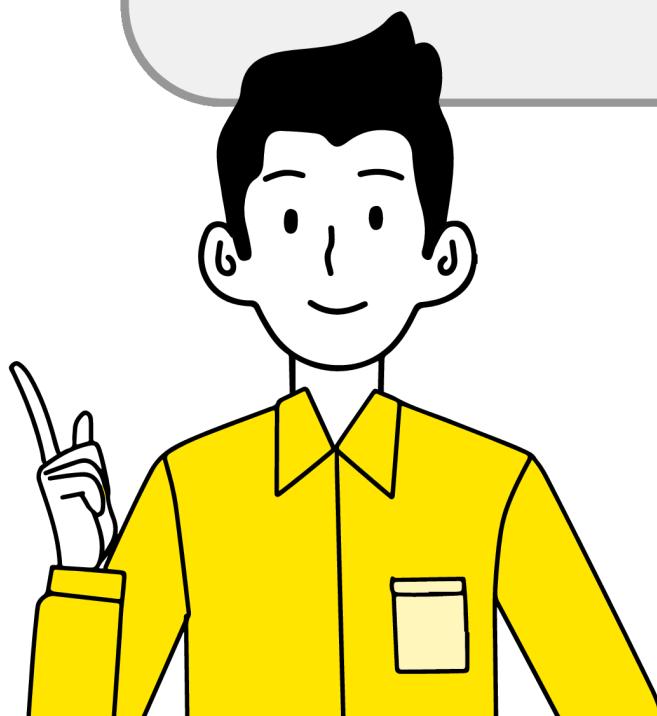
Evolution de la métrique IoU



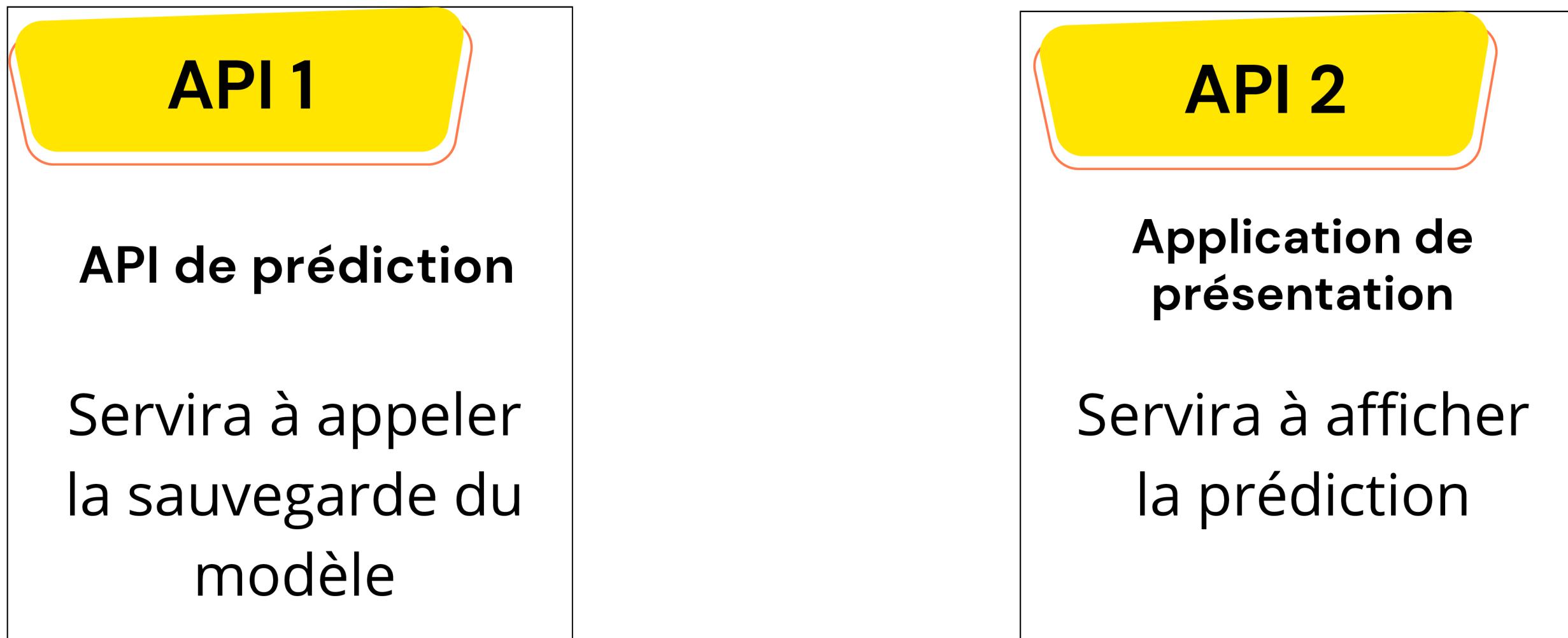
Modèle Paramètre	Unet		DeeplabV3	
Data Augmentation	Avec	Sans	Avec	Sans
Learning Rate	0,0001	0,0001	0,0001	0,0001
Taille d'images	256x256	256x256	256x256	256x256
Val IoU	0.64	0.56	0.61	0.59
Temps d'entraînement	985min	680min	735min	460min

## **Partie 3: Déploiement et Test.**

**Nous allons mettre en place deux  
API différentes.**



# Partie 3: Déploiement et Test.

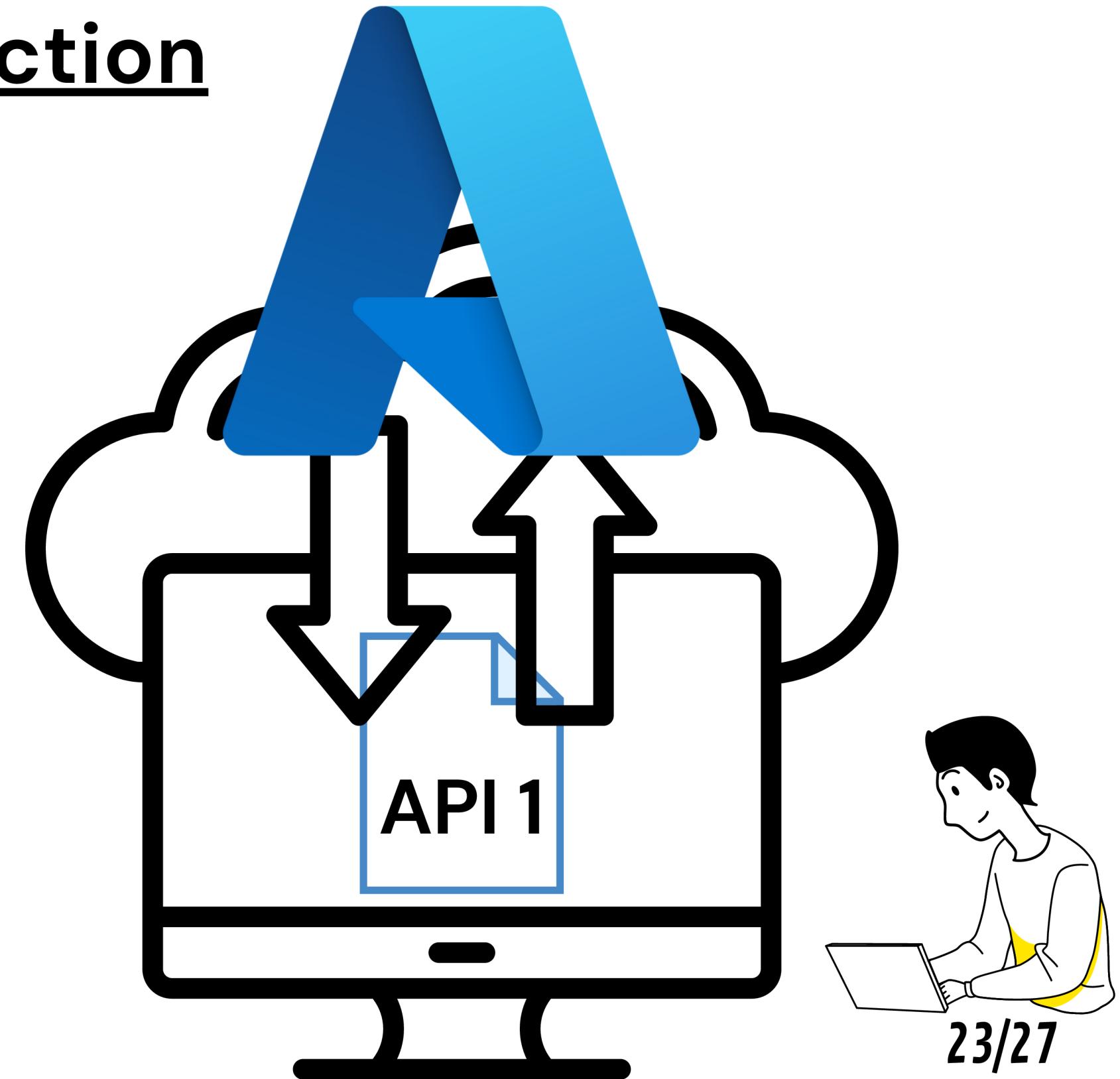


## Partie 3: Déploiement et Test.

API 1

### API de prédiction

C'est un script qui va recevoir une image et réaliser la prédiction en chargeant une sauvegarde de modèle stockée sur Azure.

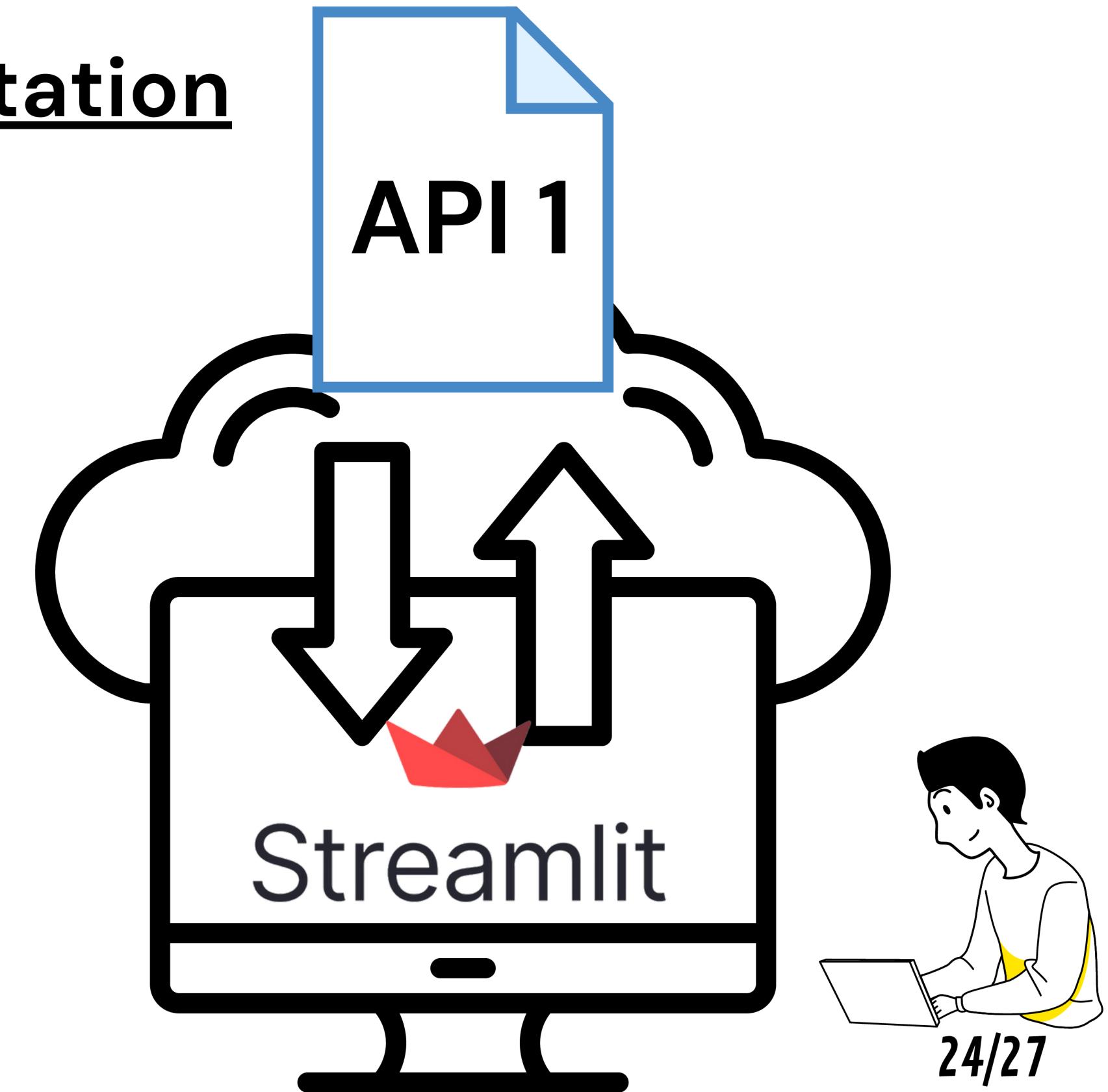


## Partie 3: Déploiement et Test.

API 2

### API de présentation

C'est un script qui va envoyer une image et charger l'API de prédiction stockée sur Azure. Pour recevoir la prediction et l'afficher.



# Partie 3: Déploiement et Test.

Exemple de résultat via l'interface



Avantage :

- Maintenance
- Optimisation des performances
- Flexibilité

## Segmentation d'images avec U-Net via Azure

Choisissez une image...

Drag and drop file here  
Limit 200MB per file • PNG, JPG, JPEG

Browse files

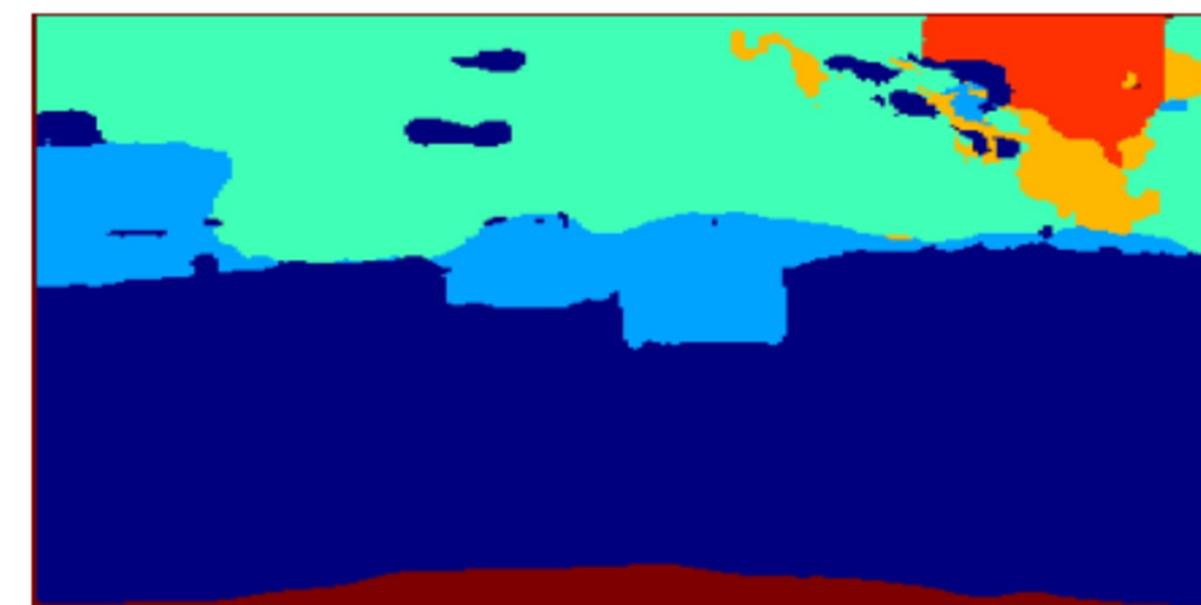
darmstadt\_000021\_000019\_leftImg8bit.png 2.2MB

x



Image chargée

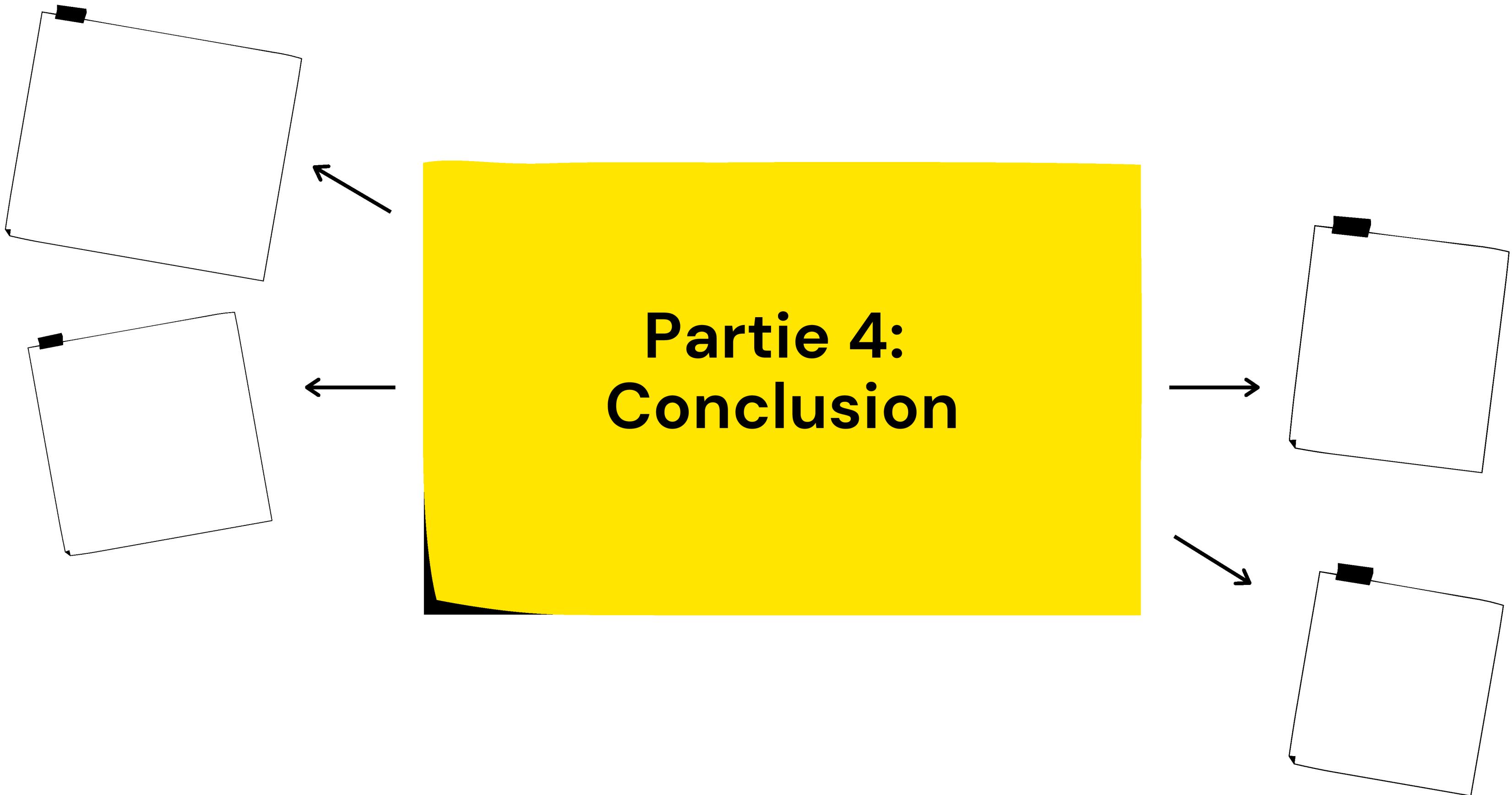
Prédiction en cours...



Masque prédictif



## **Partie 4: Conclusion**



## Partie 4 Conclusion

- Dans mes conditions, U-Net mini est le modèle le plus performant, mais demande de nombreuses heures d'entraînements, ce qui est un problème.
- Trouver un équilibre entre les nombreux paramètres et le temps. Un petit changement augmente le temps d'entraînement fortement
- Définir notre minimum de Val IoU nécessaire pour la suite du projet.



# Merci!

