FUTURE VISION TRANSPORT

DESIGN AN AUTONOMOUS VEHICLE

Contexte

Enjeux

« Future Vision Transport cherche a mettre en place un systeme de guidage pour les vehicules autonome»

Pour cela, l'equipe IA doit:

1. **Acquerir et pre traiter les images d'une camera embarque**
2. **Reconnaitre les categorie d'objets presents dans les images**
3. **Decider** les ordres de guidage a envoyer au vehicule

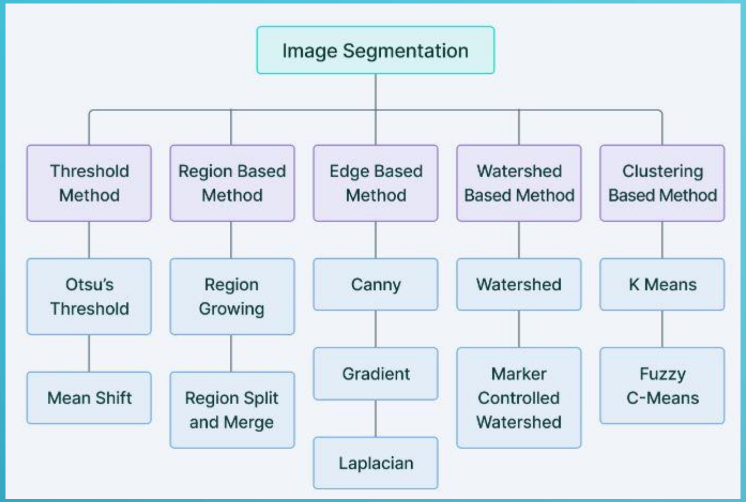
Objectifs

Nous nous interessons a l'etape 2:
Segmentation semnatique de pixels

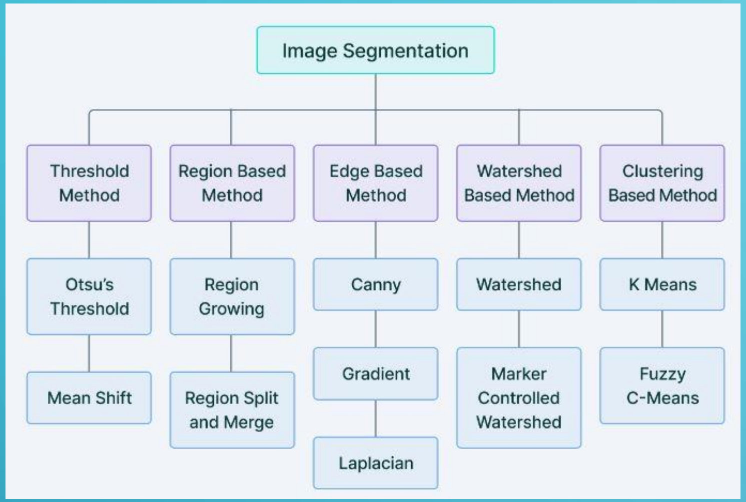
Nous allos ini chercher a tester et comparer differents modeles d'apprendissage profond (deep learning):

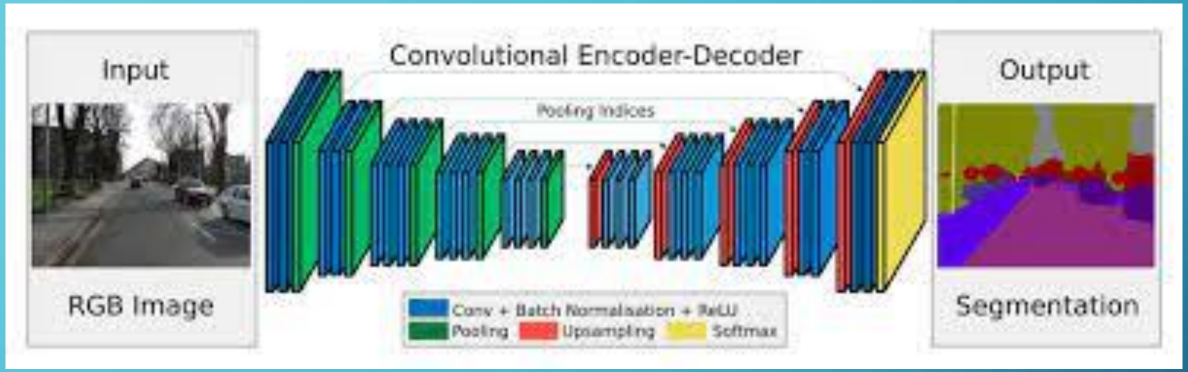
- **Architecture** du modele
- **Augmentation** des images
- **Deploiment en production (Azure)** du meilleur modele

State of the Art

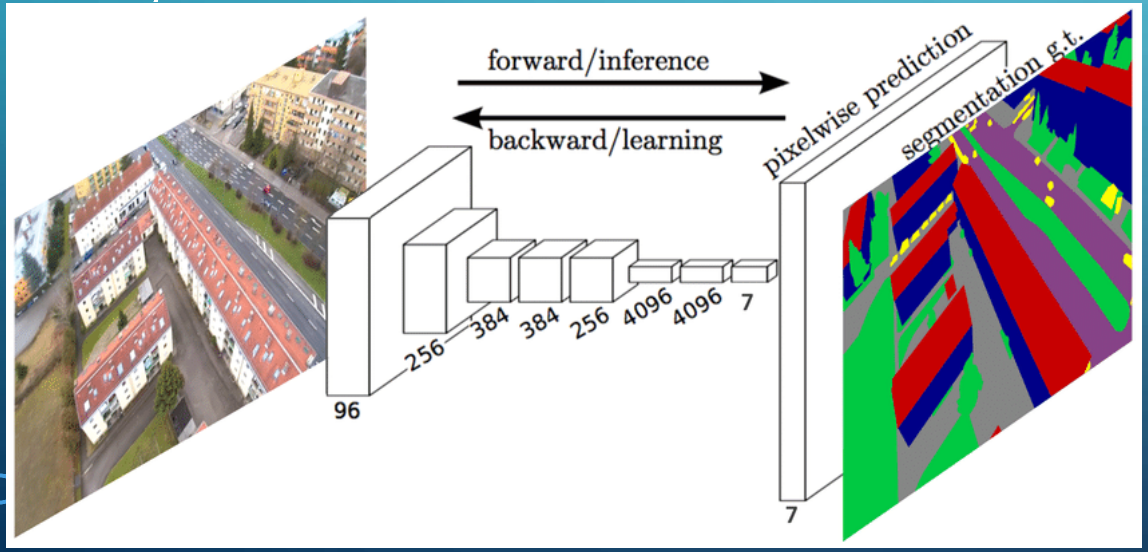


State of the Art

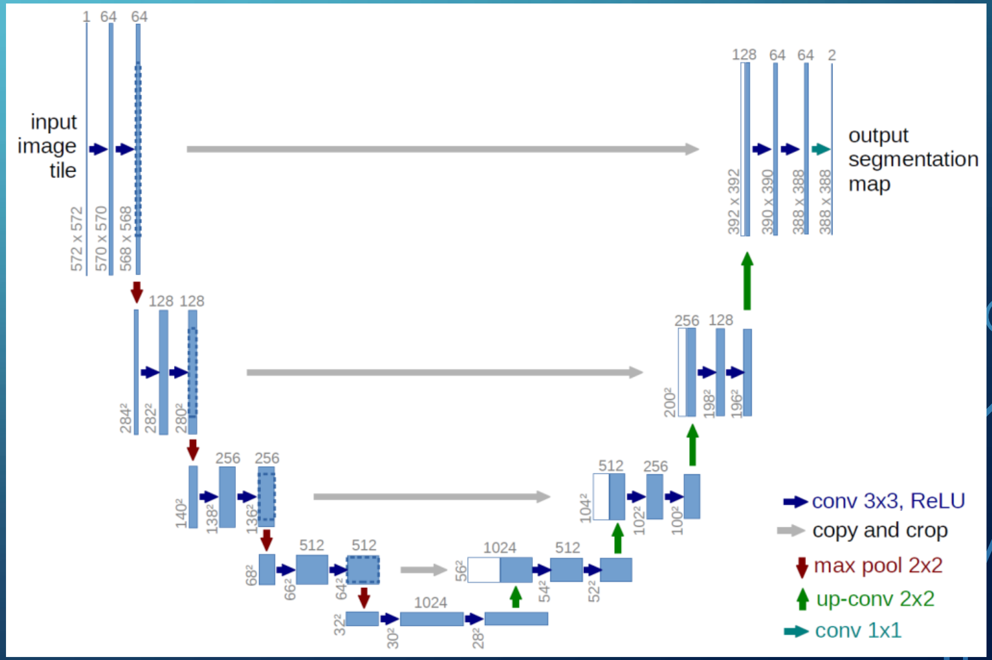




Fully Convolutional Network

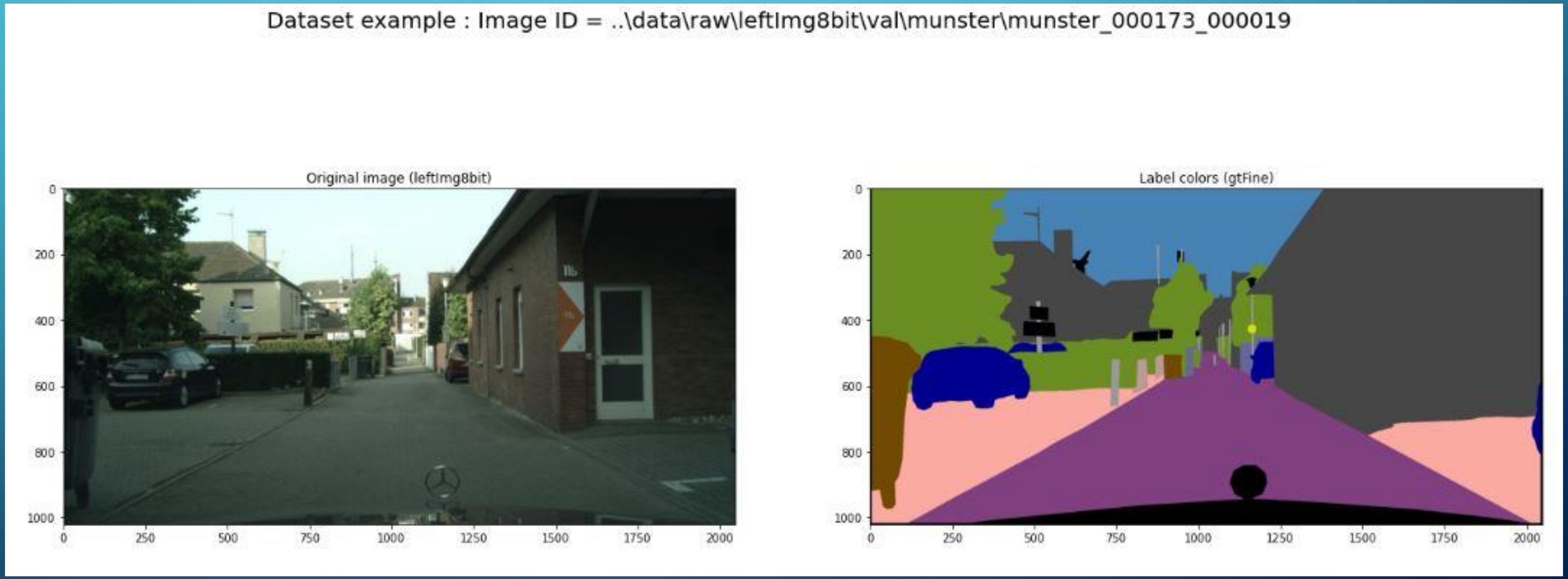


U-net



EDA: images

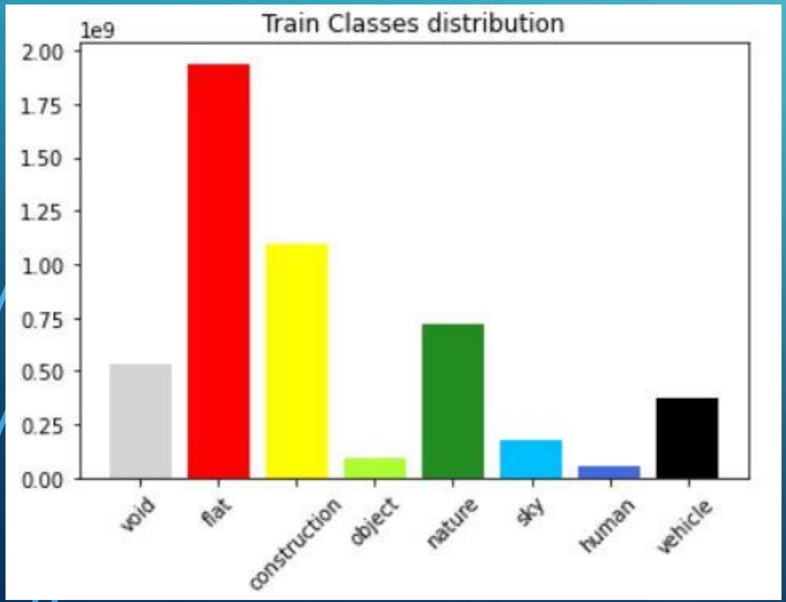
- 2975 **train images**
- 595 **validation images**
- 500 **test images**
- Size: 2048 x 1024
- Format: RGB

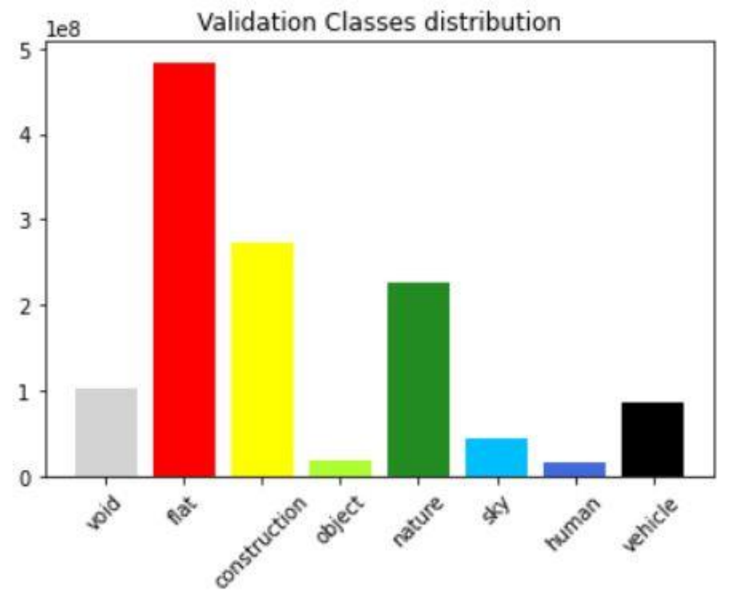


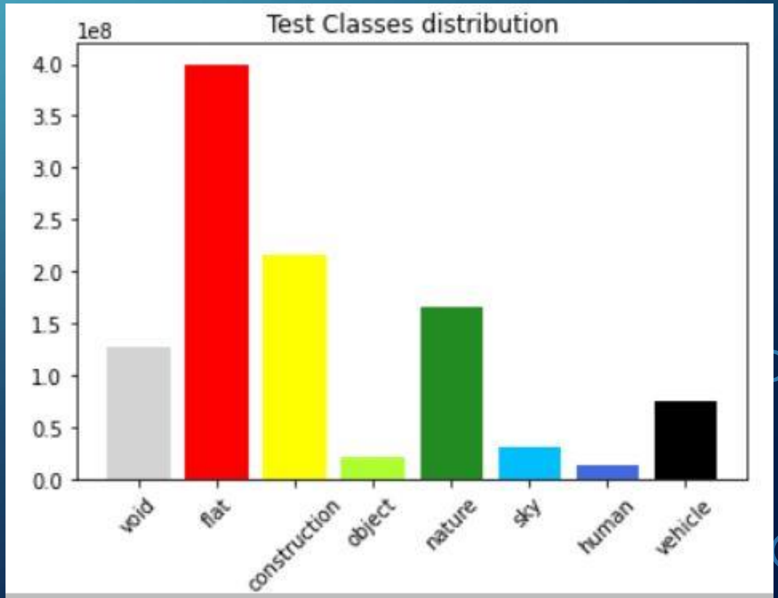
EDA: labels

8 label categories:
“void”, “flat”, “construction”, “object”, “nature”, “sky”, “human”, “Vehicle”

```
labels = [
    #name                      id    catId   category
    Label(  'unlabeled'           ,  0 ,     0 , 'void'),
    Label(  'ego vehicle'         ,  1 ,     0 , 'void'),
    Label(  'rectification border' ,  2 ,     0 , 'void'),
    Label(  'out of roi'          ,  3 ,     0 , 'void'),
    Label(  'static'              ,  4 ,     0 , 'void'),
    Label(  'dynamic'             ,  5 ,     0 , 'void'),
    Label(  'ground'              ,  6 ,     0 , 'void'),
    Label(  'road'                ,  7 ,     1 , 'flat'),
    Label(  'sidewalk'            ,  8 ,     1 , 'flat'),
    Label(  'parking'             ,  9 ,     1 , 'flat'),
    Label(  'rail track'          , 10 ,     1 , 'flat'),
    Label(  'building'            , 11 ,     2 , 'construction'),
    Label(  'wall'                , 12 ,     2 , 'construction'),
    Label(  'fence'               , 13 ,     2 , 'construction'),
    Label(  'guard rail'          , 14 ,     2 , 'construction'),
    Label(  'bridge'              , 15 ,     2 , 'construction'),
    Label(  'tunnel'              , 16 ,     2 , 'construction'),
    Label(  'pole'                , 17 ,     3 , 'object'),
    Label(  'polegroup'           , 18 ,     3 , 'object'),
    Label(  'traffic light'       , 19 ,     3 , 'object'),
    Label(  'traffic sign'        , 20 ,     3 , 'object'),
    Label(  'vegetation'          , 21 ,     4 , 'nature'),
    Label(  'terrain'             , 22 ,     4 , 'nature'),
    Label(  'sky'                 , 23 ,     5 , 'sky'),
    Label(  'person'              , 24 ,     6 , 'human'),
    Label(  'rider'               , 25 ,     6 , 'human'),
    Label(  'car'                 , 26 ,     7 , 'vehicle'),
    Label(  'truck'               , 27 ,     7 , 'vehicle'),
    Label(  'bus'                 , 28 ,     7 , 'vehicle'),
    Label(  'caravan'             , 29 ,     7 , 'vehicle'),
    Label(  'trailer'             , 30 ,     7 , 'vehicle'),
    Label(  'train'               , 31 ,     7 , 'vehicle'),
    Label(  'motorcycle'          , 32 ,     7 , 'vehicle'),
    Label(  'bicycle'             , 33 ,     7 , 'vehicle'),
    Label(  'license plate'       , -1 ,     7 , 'vehicle')]
```







MODEL SELECTION: AUGMENTATION

On a teste: Image augmention

Original

Blur

Flip

Rain

Snowflakes

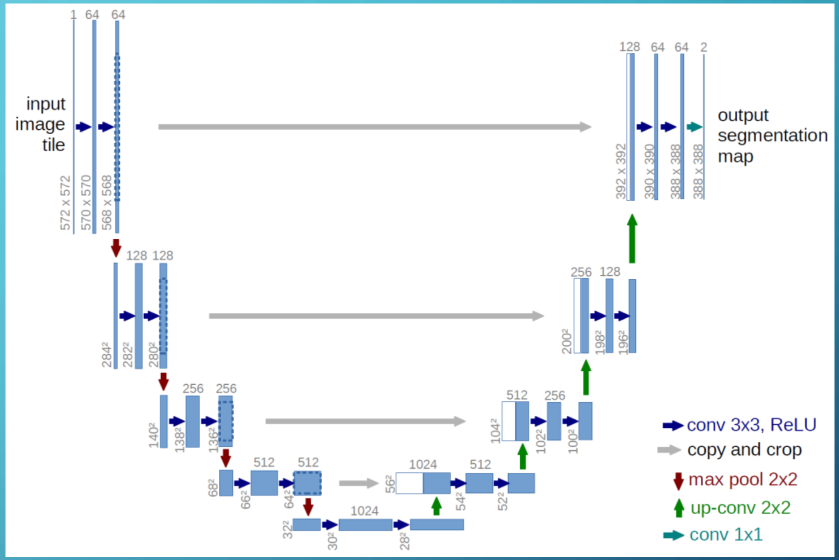
Data Generator

- Class sequence
  - __len__
  - on_epoch_end
  - _getiment__

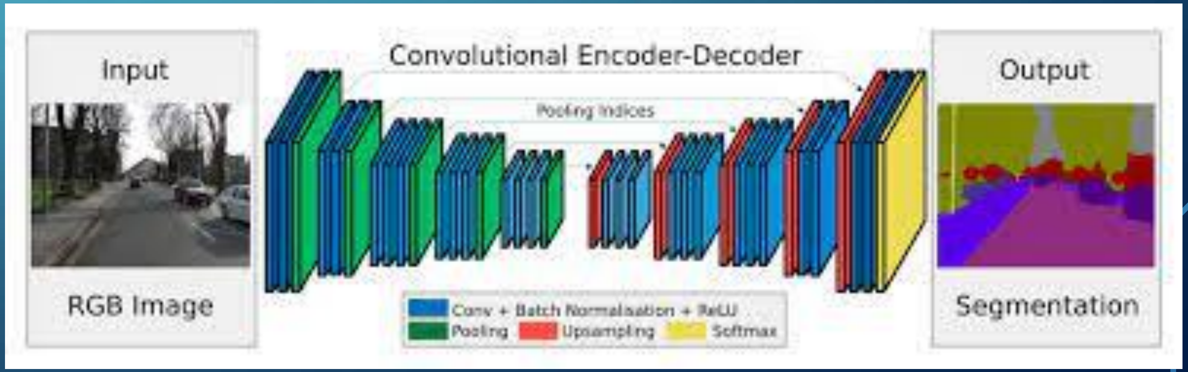
MODEL SELECTION: ARCHITECTURES

On a teste deux architecture different

- **U-Net from scratch**
  2.056.648 parameters



- **Segnet- mobilenet**
  5.528.200 parameters



LOSS AND METRICS

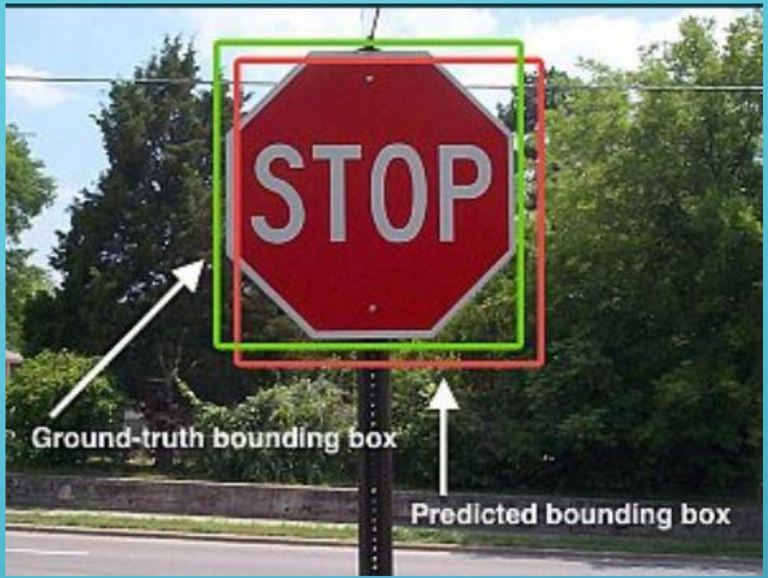
On a utilise:

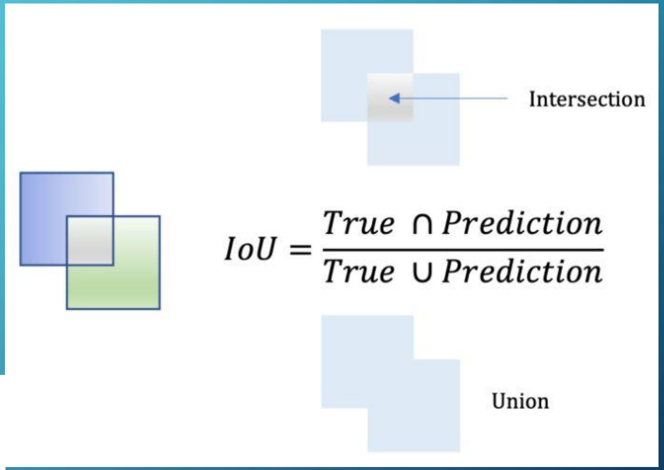
- Cross- entropy
- Jaccard Loss
- Mean Intersection over Union (mIoU) metric

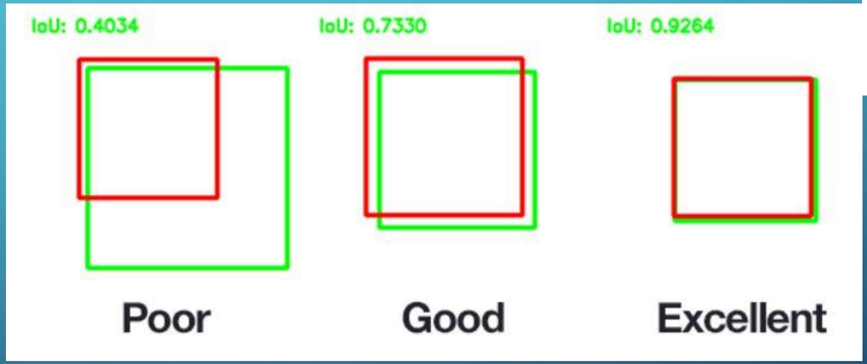
LOSS AND METRICS

On a utilise:

- Cross- entropy
- Jaccard Loss
- Mean Intersection over Union (mIoU) metric



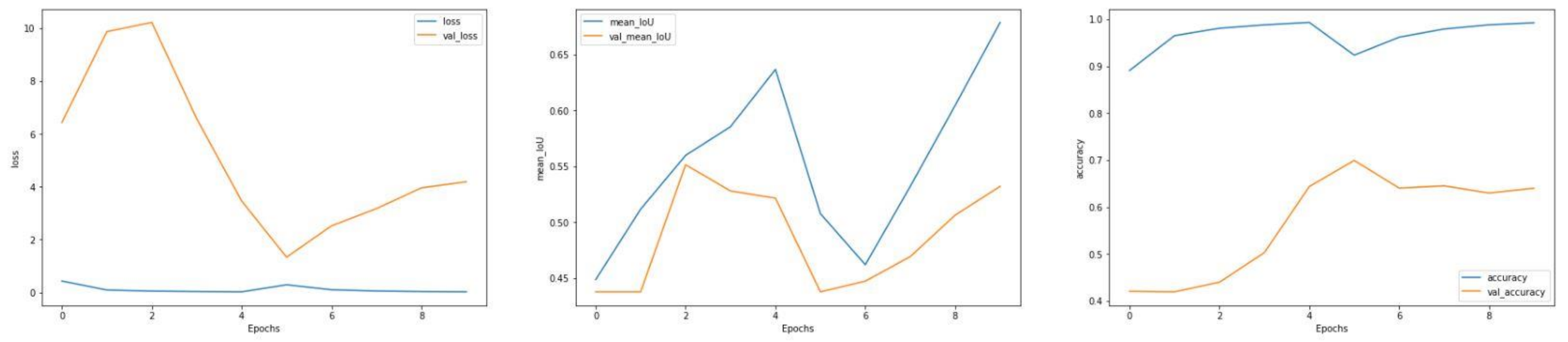


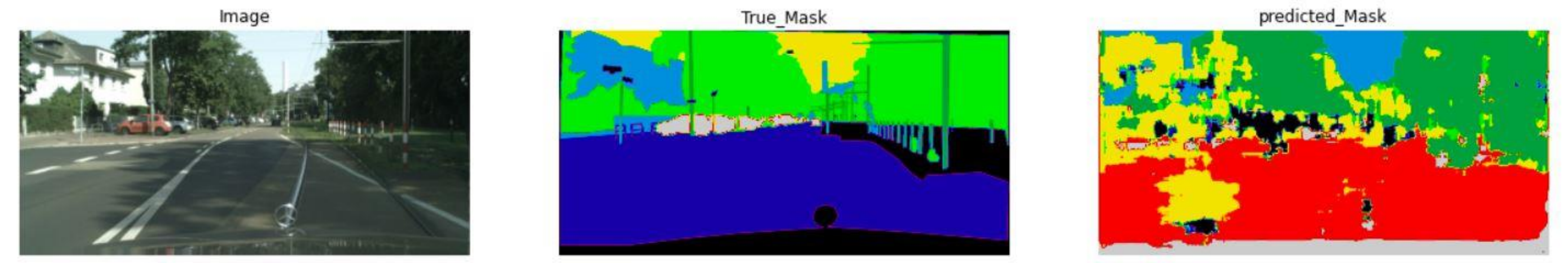


Loss	Metric	Intersection vs. Union	Confusion Matrix	Pros	Cons
Sparse Categorical Cross-entropy	Pixel Accuracy	–	$\frac{TP+TN}{TP+FP+TN+FN}$	Easy to interpret	Bad with imbalanced target classes.
Dice	F1	$\frac{2\|A\cap B\|}{\|A\|+\|B\|}$	$\frac{2TP}{2TP+FP+FN}$	Good with imbalanced target classes.	Not easy to interpret.
Jaccard	Intersection over Union (IoU)	$\frac{\|A\cap B\|}{\|A\cup B\|}$	$\frac{TP}{TP+FP+FN}$	Easy to interpret. Good with imbalanced target classes.	

RESULTS OF MODEL TESTING

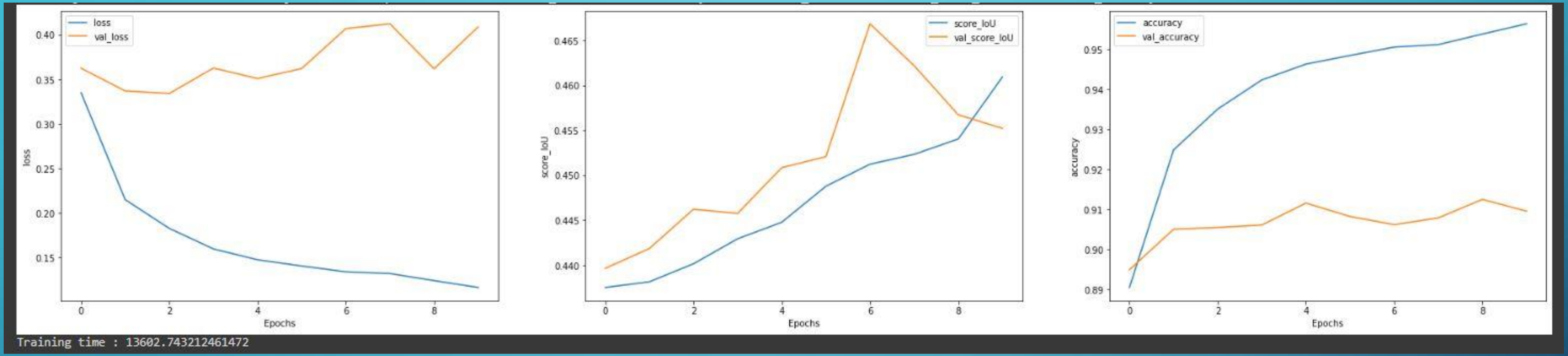
U-Net from scratch

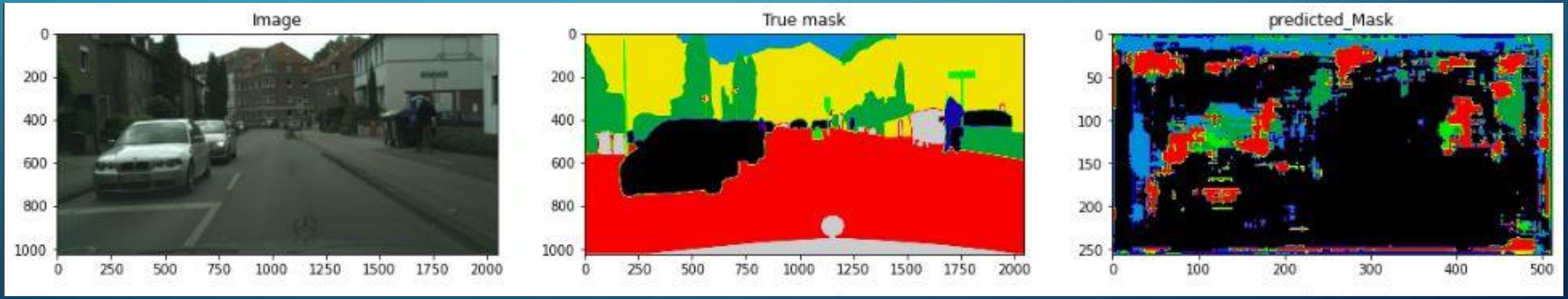




RESULTS OF MODEL TESTING

Mobilenet-segnet

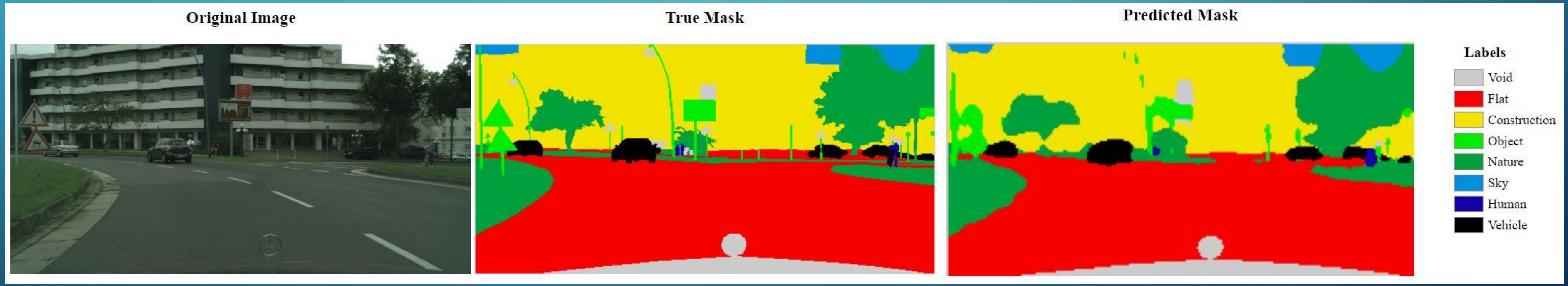




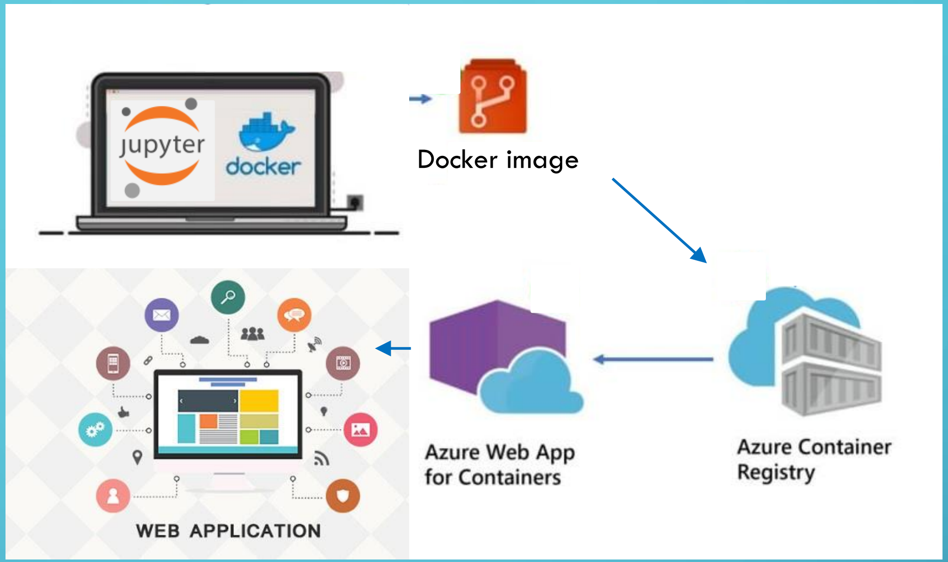
Flask Api

Nous déployons notre meilleur modèle comme une application web dans notre machine locale :

Model: Mobilenet-segnet
Augmentation: False



Web App



Nous déployons notre meilleur modèle comme une application web en Azure:

Model: Mobilenet-segnet
URL: https://esflaskapp.azurewebsites.net

CONCLUSION

Nous avons été en mesure d'évaluer les performances de différents modèles de segmentation d’images, avec différents paramètres, et de déployer le meilleur modèle en production.

Prochaines étapes....

Solutions :

Entraîner notre modèle sur beaucoup plus d'images afin d'améliorer la métrique et optimizer les differentes parametrers d’optimization.