



PRESENTATION DU PROJET

**Segmentation sémantique des photos
des élèves de 3ème année ISEM**


Elaboré par:

-Anass Mezroui
-Mustapha Bouamama

Supervisé par:

Pr. Aissam Berrahou

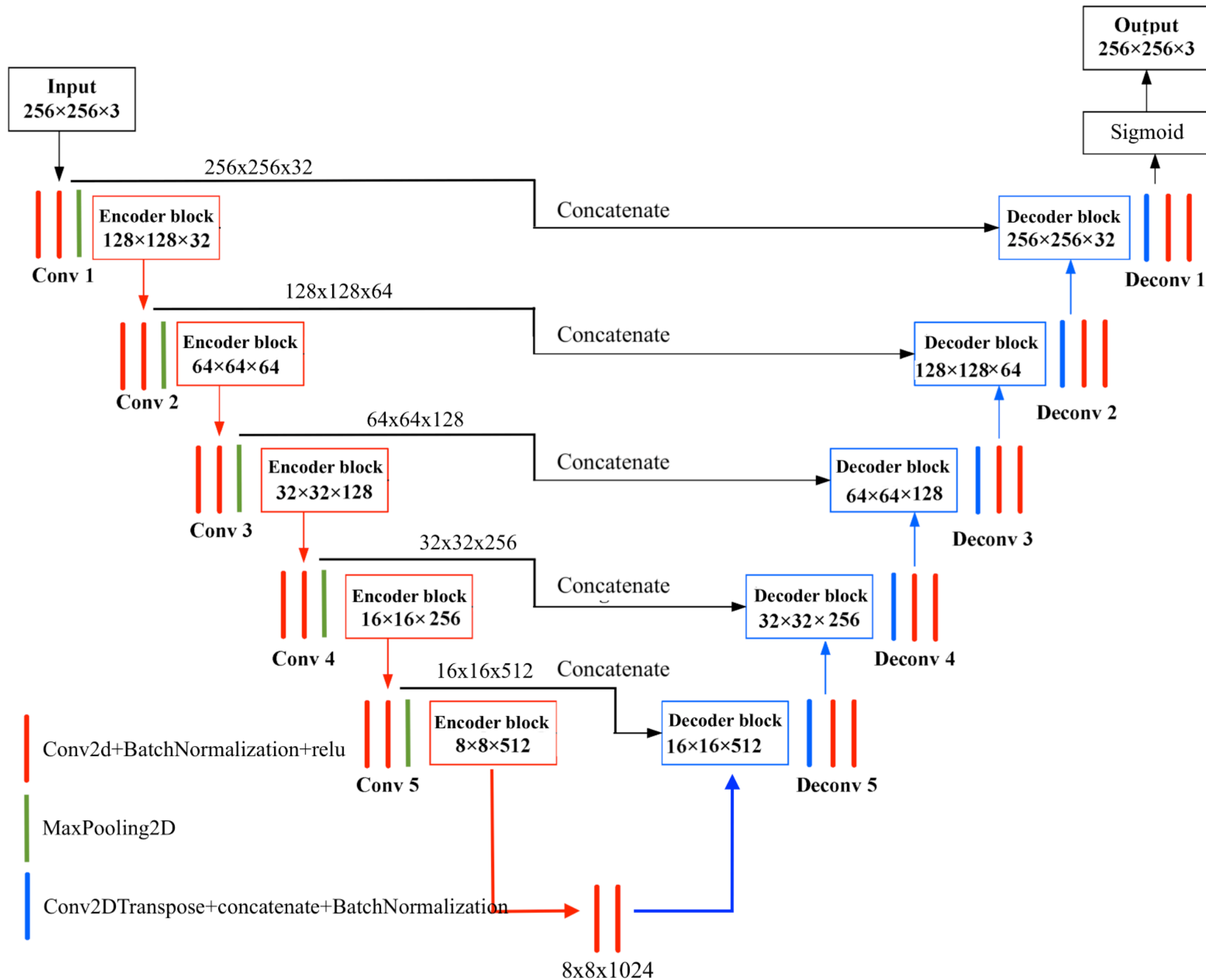
Plan:

- 
- * Exigences et context du projet
 - * Architecture de l'auto-encoder U-Net
 - * Realisation
 - * Production
 - * Conclusion

1-Exigences et context du projet

- * Le but du projet est d'entraîner un modèle auto-encodeur pour la segmentation sémantique des photos des élèves de 3ème année ISEM
- * La segmentation sémantique consiste à classer chaque pixel d'une image et lui attribuer un label
- * Notre modèle classe trois classes qui correspondent à trois élèves de la promo
- * Il existe différentes approches: auto-encodeur à base de réseaux de neurone, SegNet, U-Net.

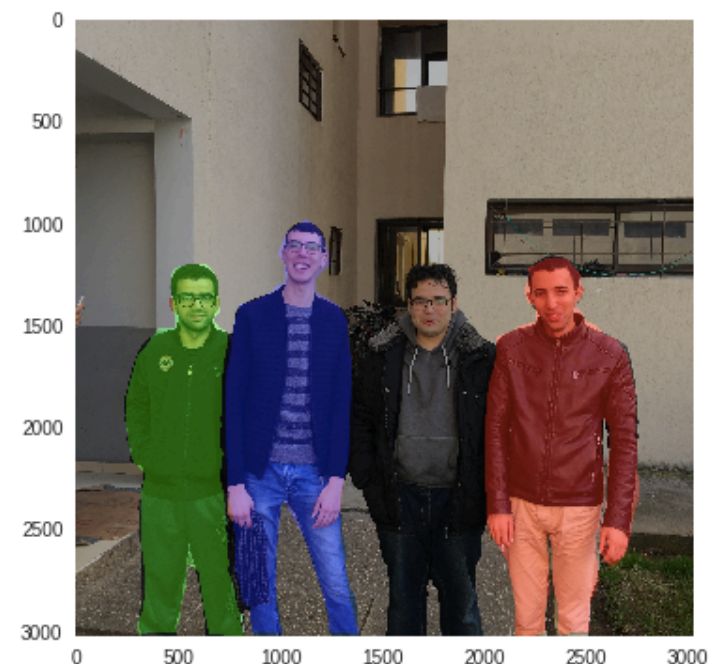
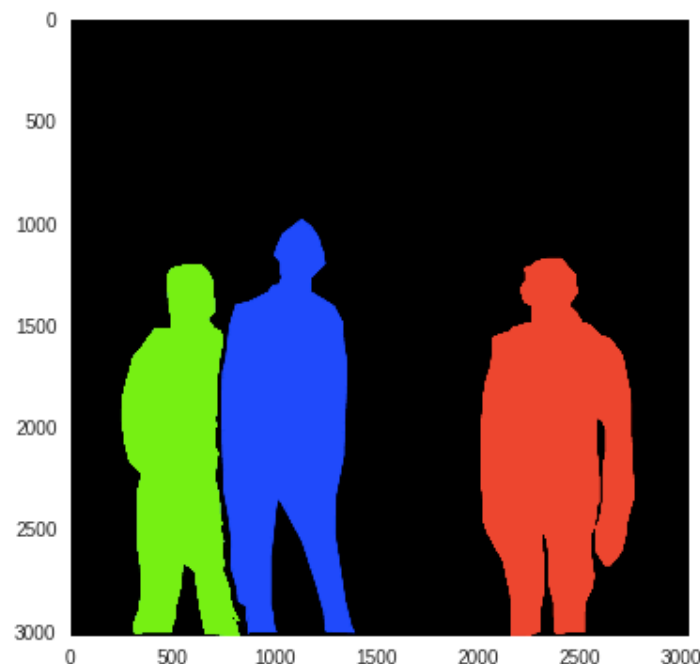
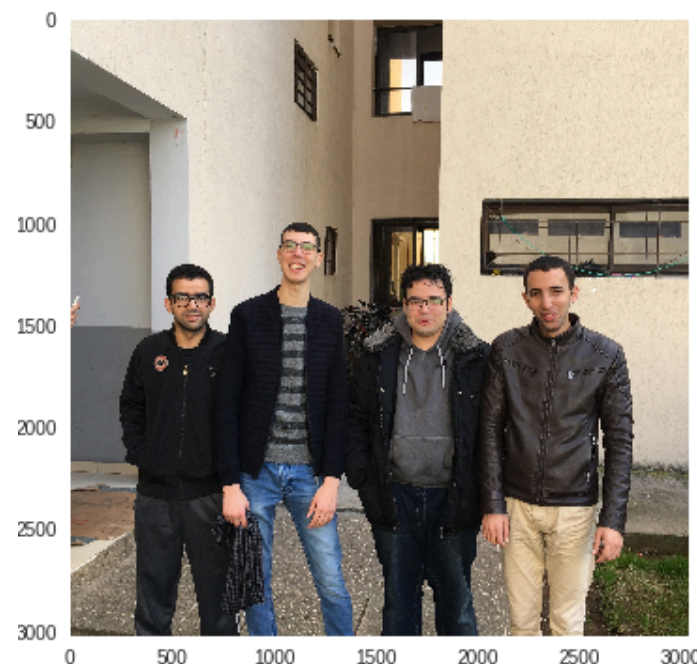
2-Architecture de l'auto-encodeur U-Net



3-Realisation

a-Creation de la dataset

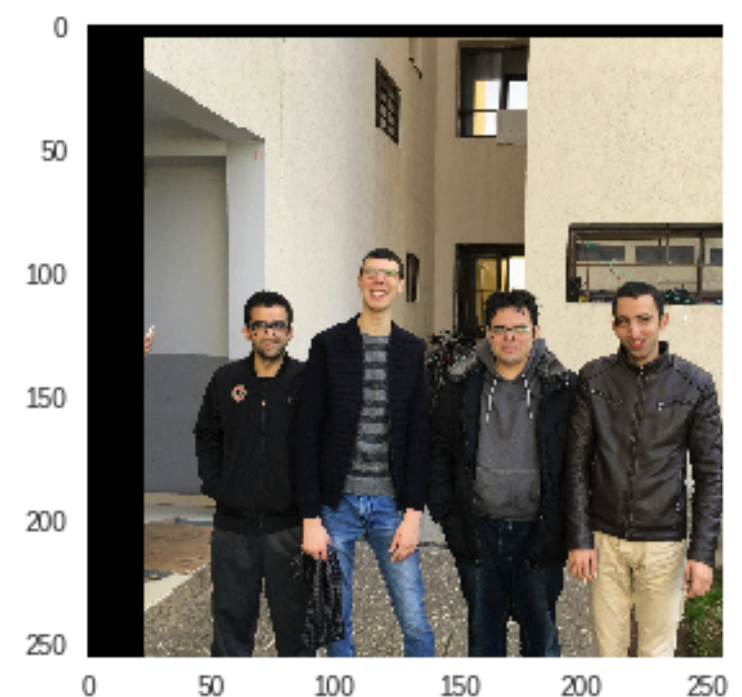
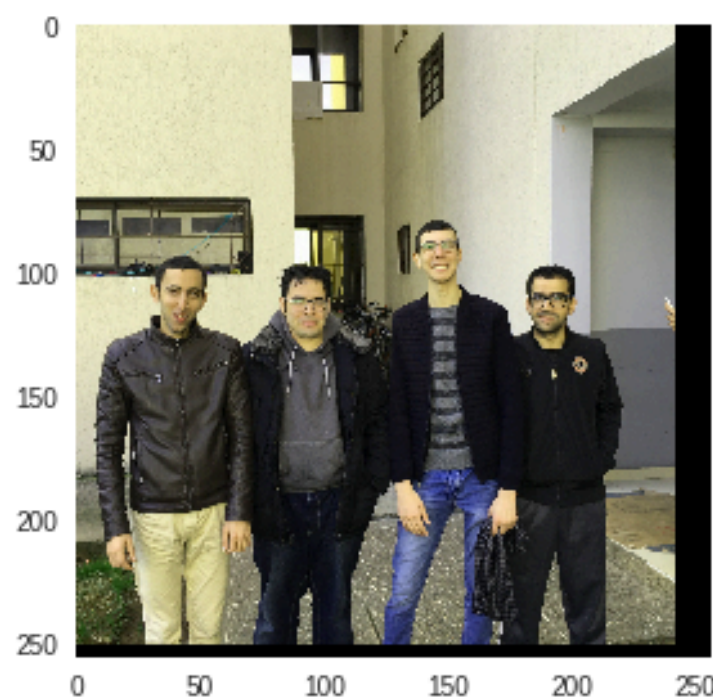
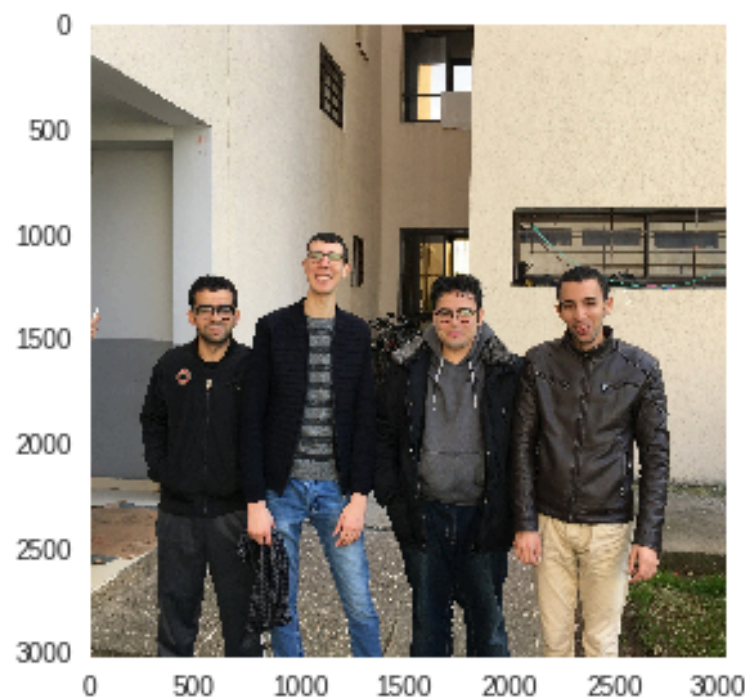
- * On a rassemblé **80** photos des élèves de la promos
- * On a utiliser l'outil **VGG Image Annotator** pour segmenter les photos manuellement, cet outil génère un fichier **csv** qui contient les coordonnées des sommets des polygones
- * On a créé les masks en utilisant **OpenCV** en se basant sur le fichier **csv**



3-Realisation

b-Creation du data pipeline

- * Le but du data pipeline est d'appliquer plusieurs transformations (redimensionnement, translation horizontale/verticale, effet miroir, hue) sur les images et les masks pour augmenter la dataset mais aussi pour réduire les effets du overfitting
- * L'avantage du data pipeline c'est que chaque epoch, on applique des transformations aléatoire sur les données ce qui fait que le modele ne revoie jamais la meme image



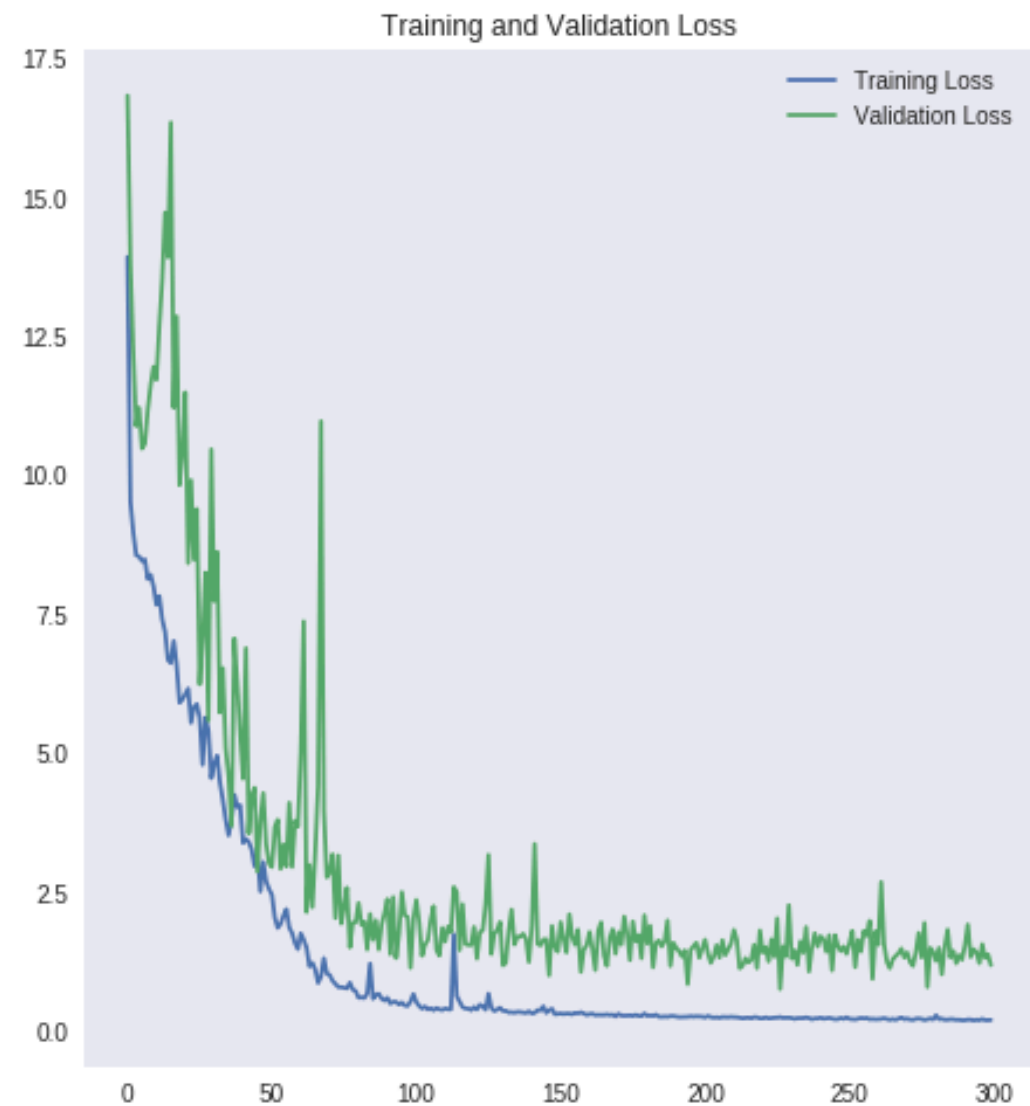
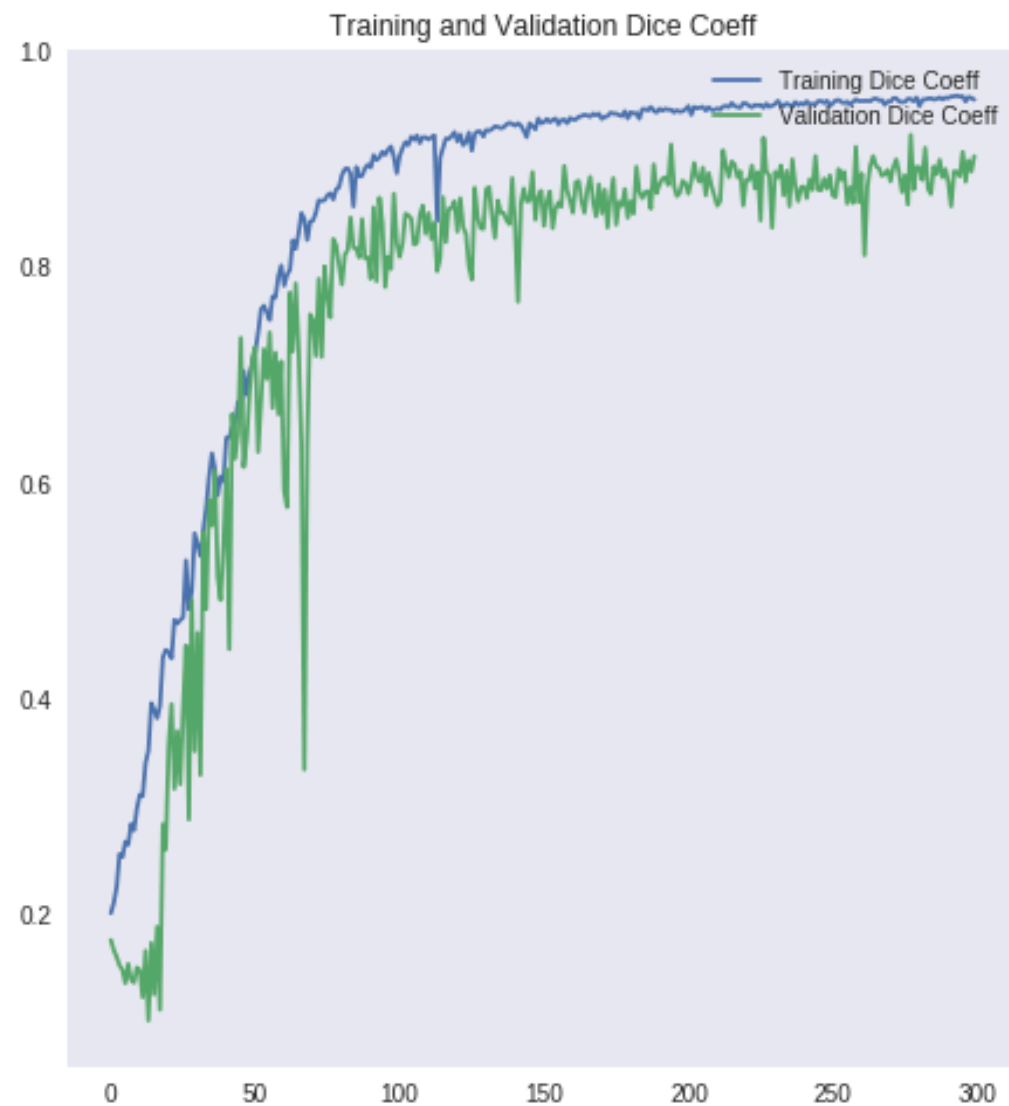
3-Realisation

c-Creation du modele

- * Pour la creation du modele on a utiliser **Keras functional API** qui est un **hight level API** qui offre une abstraction sur les detaillles des poids et des biaiss
- * On a utilisé le **dice_coefficient** comme métrique, c'est une mesure qui représente l'intersection entre deux masks
$$\frac{2 * |X \cap Y|}{|X| + |Y|}$$
- * Pour la fonction d'erreur ont utiliser **MSE**
$$\frac{1}{n} \sum_{i=1}^n (X_i - Y_i)^2$$
- * Learning rate: **0.005**, batch size: **3**
- * Nombre d'epoch: **300**

3-Realisation

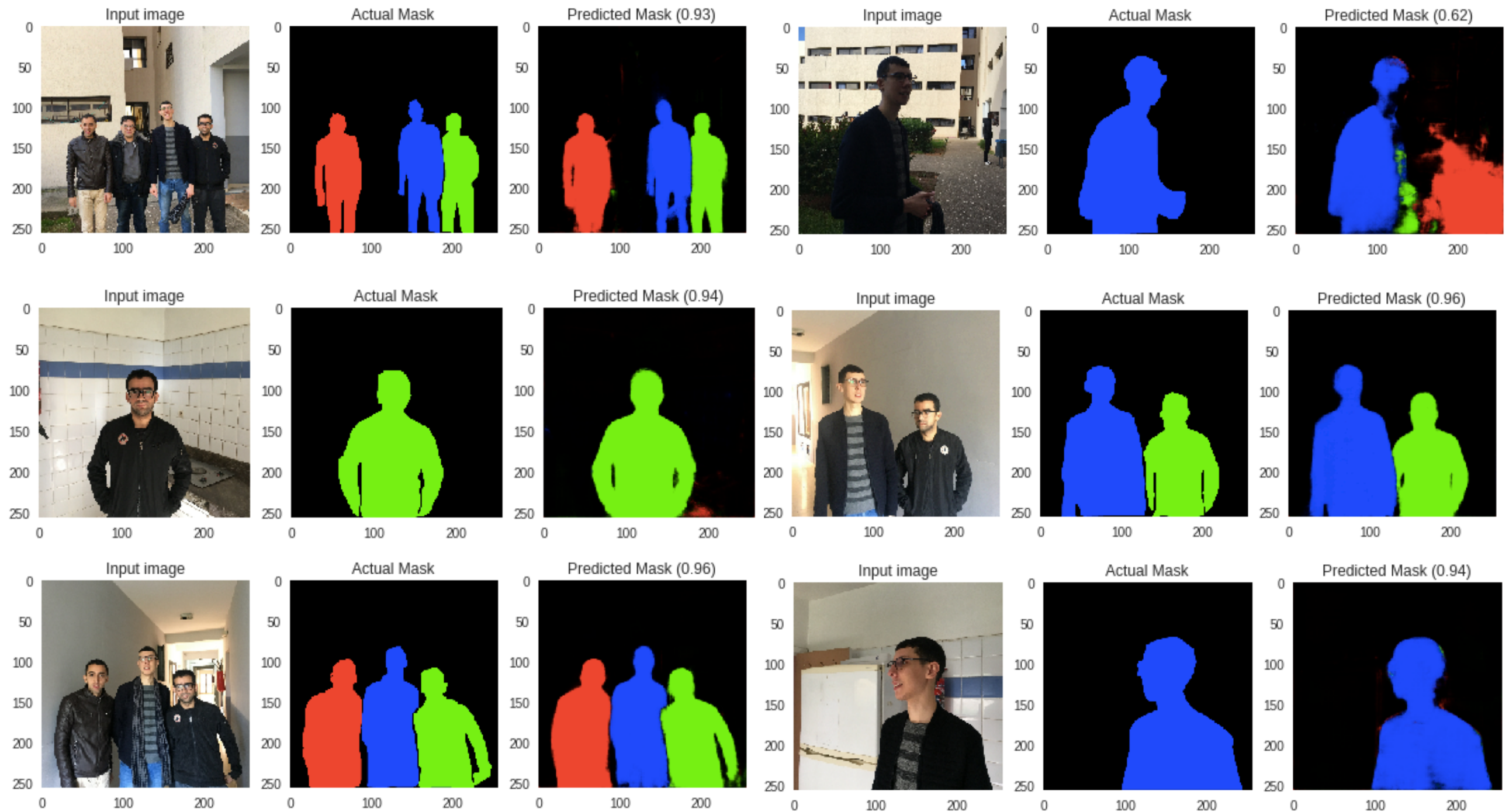
d-Entrainement et Evaluation du modele



- * Le modele a atteint **90%** pour le ***dice_coefficient*** et **1.5** pour l'erreur

3-Realisation

d-Entrainement et Evaluation du modele



4-Production

- * Deployer le modele pour la production sur **tensorflow.js** et proposer une application web qui permet de charger une image et de la segmenter dans le navigateur.



Choose File test_img__.JPG

Segment

Image



Mask



4-Conclusion

- * Le modele utilisé pour segmenter les photos des élèves de 3eme année ISEM promos 2019 arrive a atteindre un taux de reconnaissance de 90%
- * Le modele pourrait être améliorer en utilisant d'autres fonctions d'erreur qui sont moins sensibles à la distribution des différentes classe de données
- * Le modele fait 125MB se qui n'est pas pratique pour l'envoyer à chaque fois vers le client, en plus il prend un temps considerable pour s'executer sur le navigateur, ce qui suggère la possibilité d'utiliser un service web pour executer le modele sur une machine distante.