



## **5A ILC Sujet de TP/Projet**

**2021- Détection de port de masque**

**Romain ANQUETIN  
Peter BACHOUR**

# 1.Introduction

Ce projet est à but pédagogique, il nous permet de voir l'ensemble de la chaîne permettant de concevoir une application à base de Machine Learning incluant des méthodes de Deep Learning

Ce type de projet fait parler le monde entier récemment surtout avec les obligations de port de masque dans des zones spécifiques. Le développement d'un système ou d'une interface qui arrive à savoir si quelqu'un porte bien son masque ou non tout en l'intégrant dans n'importe quel système de télésurveillance.

Le projet porte donc sur 3 axes principaux, nous allons commencer par la conception d'une base d'image annotées destinée à l'entraînement d'un modèle de Machine Learning. Ensuite, nous choisirons un modèle adapté sur lequel nous effectuerons l'entraînement. Enfin, la conception d'une interface nous permettant de visualiser en temps réel la détection de masque.

## 2.La récolte de données

Afin d'avoir un jeu de données assez consistant, une contribution équitable de chaque étudiant de la promotion a eu lieu. Chaque étudiant a ajouté environ 100 images. Ceci nous a permis de créer un dataset d'image annotées d'un total de 1755 images unique. Toutes les images ont été téléchargées sur un dépôt GitHub commun à tout le monde :

<https://github.com/jolignier/face-mask-dataset-ilc-2021>.

Nous nous sommes tous mis d'accord sur la même annotation des images, ces images ont été divisées en 3 catégories :

- "**with\_mask**" : un masque correctement porté qui recouvre le nez et la bouche
- "**with\_incorrect\_mask**" : un masque porté sous le nez, ou de façon pas très covid-friendly (sur les yeux, sous le menton ...)
- "**without\_mask**" : un visage sans masque

Afin d'éviter tout doublon dans notre base d'image, nous avons suivi un processus d'ajout d'image. Ce dernier consiste à :

- Le nom du fichier image doit correspondre uniquement au hash md5 du contenu du fichier.
- Toutes les images modifiées en mode Photoshop seront interdites dans la base d'images.
- Les doublons ont été supprimés grâce à l'aide d'un script qui compare la similitude entre toutes les images du dataset.

Afin que l'annotation soit similaire chez tout le monde et pour ne pas avoir des différences dans notre dataset, nous avons pris la décision d'utiliser l'outil « LabelImg » dont l'interface est visible en figure 1 ci-dessous.

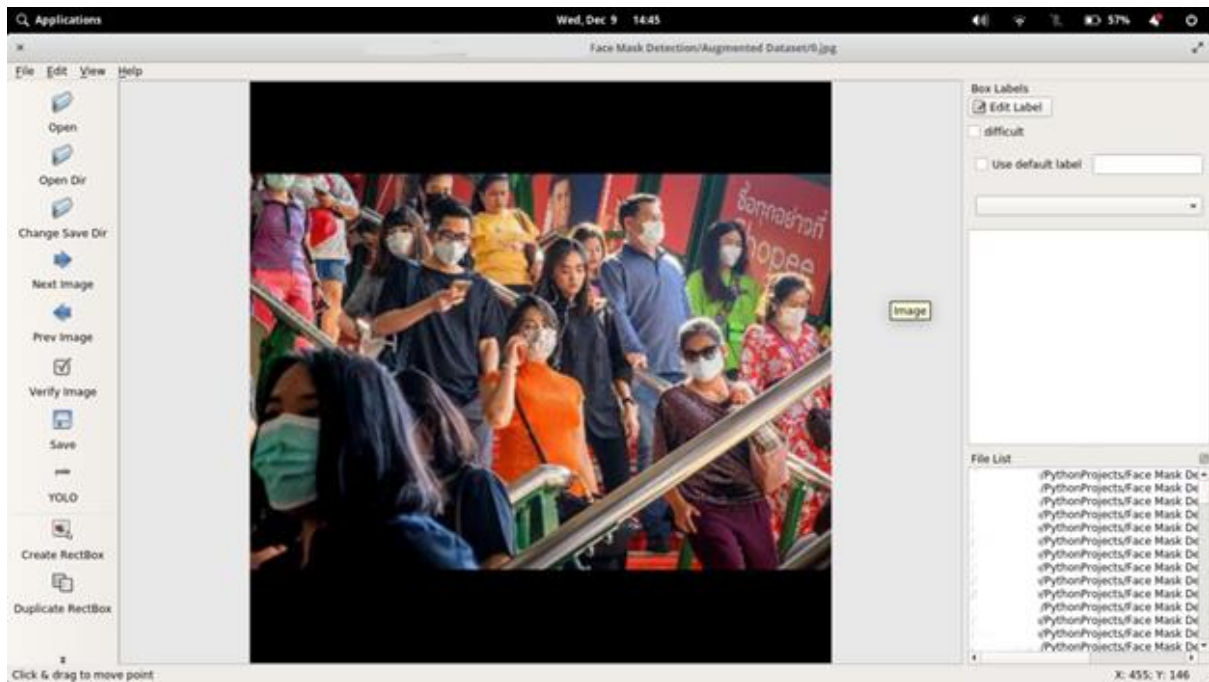


Figure 1 - Interface du logiciel Labellmg

Ce logiciel gratuit et libre d'accès permet de générer des annotations aux format PascalVOC qui est demandé par l'enseignant, et très utilisé en Machine Learning.

Pour augmenter le nombre de donnée nous avons utilisé le dataset crée par la promotion précédente.

Les visages sont annotés selon les trois classes citées précédemment (**with\_mask**, **with\_incorrect\_mask**, **without\_mask**) et de sorte à encadrer le visage depuis le milieu du front jusqu'au bas du masque/menton afin de correspondre au jeu de données de test imposé par AndrewMvd sur Kaggle: <https://www.kaggle.com/andrewmvd/face-mask-detection>.

### 3.Choix du modèle de réseau de neurones

Le but de ce projet est de détecter si une personne a bien mis son masque ou non. Par suite, il est évident que nous allons nous orienter vers un modèle de réseau qui détecte les objets et en plus particulier les visages.

Après avoir étudié les différents modèles existants et tout en basant notre recherche sur des modèles de détecteur d'objet. Nous nous sommes rendu compte qu'il est impérativement nécessaire d'ajouter une nouvelle contrainte pour notre recherche de modèle. Notre modèle doit être capable de détecter un grand nombre d'objets dans une image en même temps parce que notre application sera utilisée dans des zones où plusieurs personnes seront potentiellement présentes.

En rétrécissant notre recherche, nous avons décidé de se pencher sur l'implémentation Bytoch du modèle YoloV5 de Ultralytics qu'on peut trouver sur le lien GitHub suivant : <https://github.com/ultralytics/yolov5> .

YOLOv5 🚀 est une famille d'architectures et de modèles de détection d'objets pré-entraînés sur l'ensemble de données COCO, et représente la recherche open source d'Ultralytics sur les futures méthodes d'IA de vision, incorporant les leçons apprises et les meilleures pratiques évoluées au cours de milliers d'heures de recherche et développement. Ce modèle permet la reconnaissance d'objet en temps réel ce qui correspond parfaitement à nos besoins.

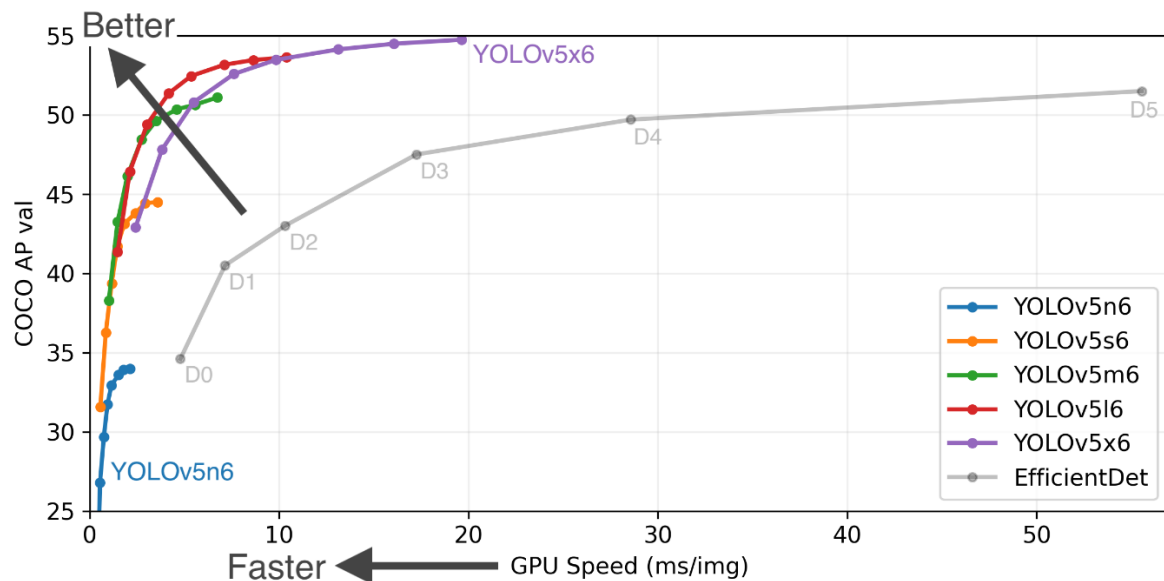


Figure 2 - Comparaison des performances entre les versions de YoloV5 et de EfficientDet

## 4. Augmentation de données et gestion du dataset

Nous utiliserons la plateforme Roboflow pour la gestion du dataset, Roboflow permet d'observer la distribution des données dans le dataset.

Roboflow nous permet aussi de faire de l'augmentation de données, nous utilisons l'augmentation de données via Roboflow et yoloV5. Les augmentations que nous avons utilisées sont :

- Rotation de l'image entre  $-15^\circ$  et  $15^\circ$ , permet de détecter les têtes penchées.
- Une rotation de  $90^\circ$  dans l'espace hue, permet de détecter correctement avec différente condition de couleur.
- Un flou gaussien de 2px, permet de détecter les visages en arrière-plan.
- Des mosaïques, cela permet de détecter les visages coupés
- Un effet miroir
- Du bruit, permet de détecter des visages incomplets

## 5. Mise en œuvre fonctionnelle

Pour l'entraînement nous avons fait 15 epochs avec une taille de batch de 20. Nous avons remarqué que le réseaux yolov5s overfeat au bout de 10 epochs. De plus, à cause de notre dataset assez faible en image malgré l'augmentation de donnée, nous avons choisi un modèle déjà entraîné.

Comme nous pouvons le voir sur l'image suivante, notre réseau a pu prédire les différentes situations de port de masque. Comme nous pouvons le voir dans la photo à gauche, le réseau a prédit que la personne porte un masque avec une certitude de 90%. En effet on visualise des rectangles autour des visages des différents individus. Ces rectangles correspondent à la prédiction de notre réseau de neurone.

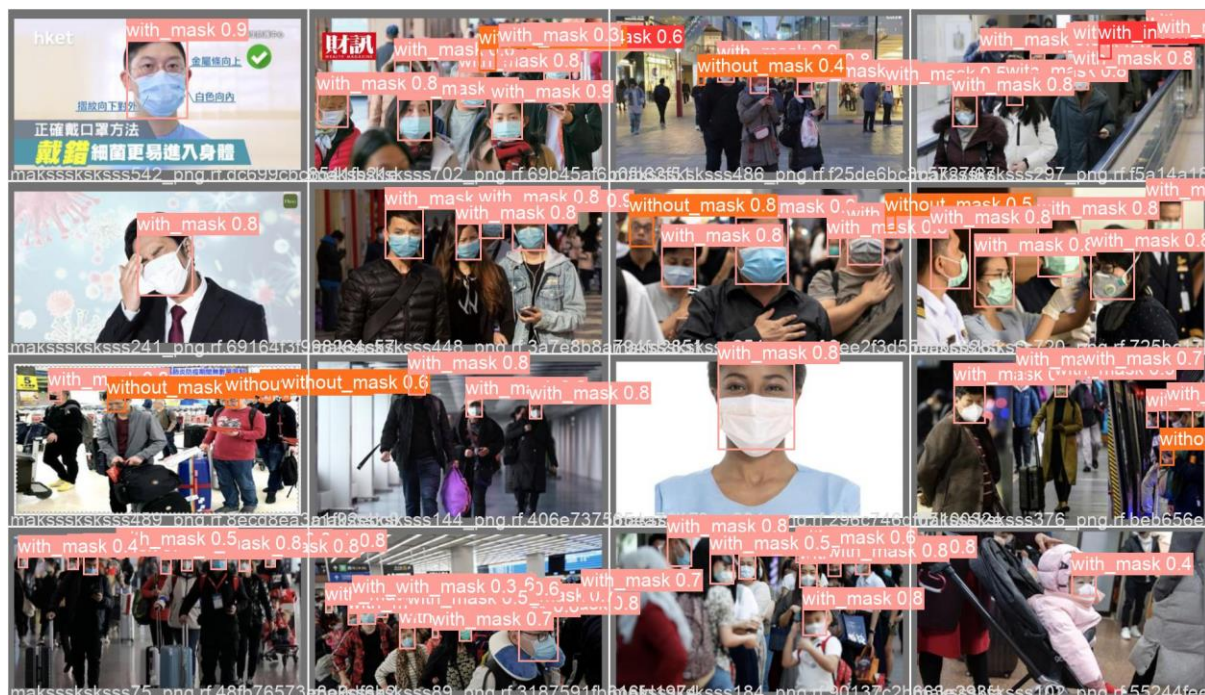


Figure 3 - Prédiction de notre réseau

## 6. Conclusion

En conclusion ce projet est grave intéressant, nous avons pu travailler sur ce dernier dès le début en choisissant notre propre modèle de réseau de neurone tout en appliquant toutes les compétences théoriques apprise auparavant.

Le projet est disponible à cette adresse : <https://github.com/Mortpo/SuperMaskDetector>