

Rapport de Projet : Détection de Casque de Chantier et de Veste Fluorescente avec YOLO, Flask et Webcam

Introduction

Dans ce rapport, je vais présenter mon projet visant à développer un modèle de détection de casques de chantier et de vestes fluorescentes, en utilisant YOLO comme modèle de détection d'objet. L'objectif final était d'intégrer ce modèle dans une application Flask, permettant la détection en temps réel via la webcam.

Collecte de Données

Pour entraîner le modèle YOLO, la première étape a consisté à constituer un ensemble de données adéquat. J'ai créé mon propre dataset en utilisant la méthode de web scraping pour collecter des images de casques de chantier et de vestes fluorescentes depuis diverses sources en ligne. J'ai ensuite annoté manuellement ces images à l'aide de MakeSenseAI.

Prétraitement des Données

Les images annotées ont été prétraitées pour être utilisées dans le modèle YOLO. J'ai utilisé Roboflow pour automatiser le processus de génération du fichier de configuration YAML nécessaire pour l'entraînement de YOLO. De plus, Roboflow a permis de diviser automatiquement le jeu de données en ensembles d'entraînement et de test, optimisant ainsi la formation du modèle.

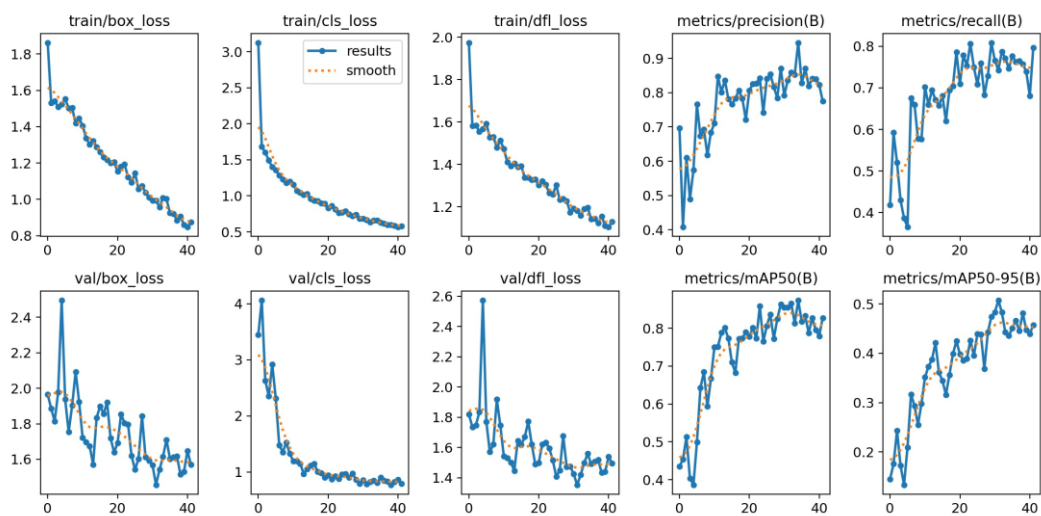
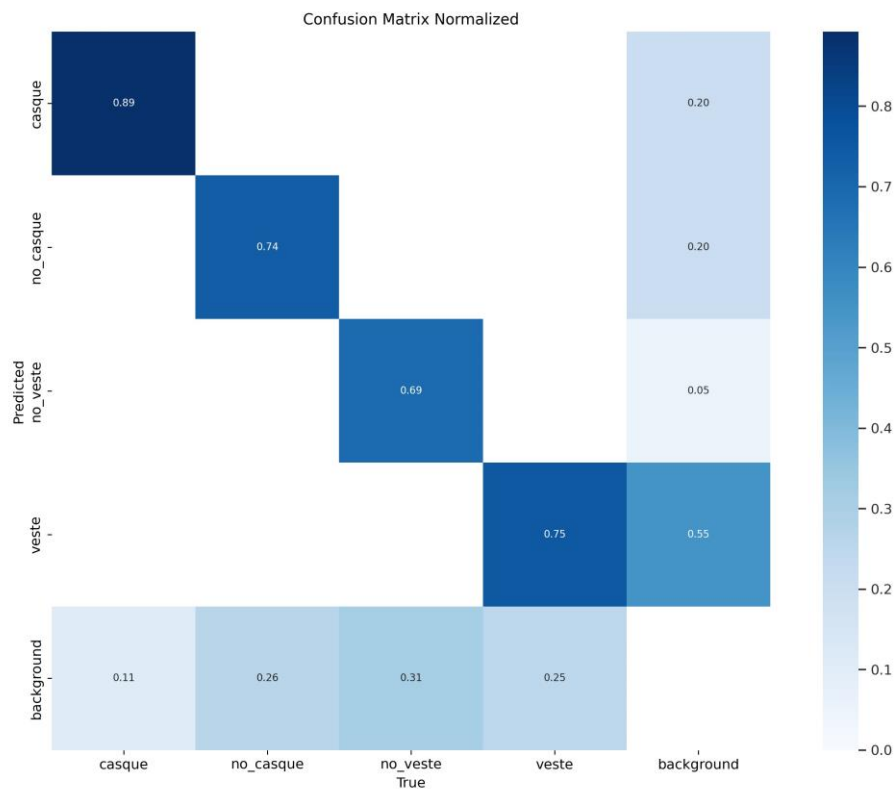
Entraînement du Modèle YOLO

Après avoir préparé les données, j'ai utilisé YoloV8 sur collab pour l'entraînement du modèle. Le snippet pour récupérer le dataset et le fichier yml est très simple grâce à roboflow, un simple copier-coller suffit.

Intégration dans une Application Flask

Une fois le modèle YOLO entraîné, J'ai intégré avec le modèle dans une application Flask. J'ai utilisé des extraits de code provenant d'une vidéo trouvée sur internet pour configurer le flux vidéo de la webcam en temps réel et pour effectuer la détection d'objets en utilisant mon modèle. Cette application a été développée sous environnement Windows car la webcam ne marche pas sous WSL.

Résultats et Évaluation



D'après la matrice de confusion, on peut voir que le modèle a plus de mal sur la catégorie "no_veste", probablement dû à la faible quantité de ce type de donnée comparé aux autres, le

dataset aussi était léger, seulement 200 images environs. Il performe correctement pour les autres catégories au vu du peu de données.

Conclusion

Ce projet a démontré avec succès la création d'un modèle YOLO pour la détection de casques de chantier et de vestes fluorescentes, ainsi que son intégration dans une application Flask permettant une détection en temps réel via la webcam.

On peut toutefois noter que le modèle a certaines limitations, notamment quand les personnes sont penchées : il a du mal à détecter.