Explication du projet – YOLOv8 Voitures (Streamlit)

Ce projet a pour objectif de démontrer l'application concrète d’un modèle de détection d’objets dans un contexte de vision par ordinateur. Il s'agit d'une interface web interactive développée avec Streamlit, qui permet à un utilisateur de téléverser une image et de détecter automatiquement les voitures présentes grâce au modèle YOLOv8. L’interface offre un retour visuel avec l’image annotée, un tableau de résultats techniques (coordonnées, score de confiance), et propose également le téléchargement du résultat annoté. Le projet illustre ainsi une chaîne complète allant de l’analyse d’une image brute jusqu’à la restitution interactive du résultat à l'utilisateur, tout en intégrant des éléments d’optimisation et une gestion des cas particuliers.

Dans la Partie 3 – Tests et manipulations, des images de test ont été incluses dans le dossier du projet pour illustrer différents cas d'utilisation. Ces tests valident la robustesse de l'application en simulant des situations réelles : une image avec plusieurs voitures, une sans aucune voiture, et une avec d'autres objets comme des personnes ou des arbres. Par exemple, lorsqu’on téléverse une image sans voiture, l’application affiche automatiquement un message d’avertissement indiquant qu’aucun objet de type voiture n’a été détecté. Lorsque plusieurs voitures sont présentes sur l’image, l’application détecte le nombre exact de voitures. Dans le cas d’une image contenant des voitures et d'autres objets comme des piétons, le modèle peut parfois détecter ces objets non demandés, bien que le filtrage soit censé n’afficher que les voitures.

Il convient de noter que sous certaines conditions, notamment avec des images floues, sombres ou mal orientées, le modèle peut se tromper ou détecter incorrectement des objets.

Pour accélérer la détection, j'ai intégré un redimensionnement automatique des images après téléversement. Cela permet de réduire la taille des images analysées, ce qui améliore la rapidité d’exécution. Cependant, cette réduction des dimensions peut parfois affecter la précision, en particulier lorsque les objets deviennent trop petits ou flous après redimensionnement. Ce facteur contribue à une baisse occasionnelle du score de confiance, voire à des prédictions incorrectes sur certaines images.

Illustration de l'effet du redimensionnement sur la détection :

Image avec redimensionnement (score plus bas, image plus petite):

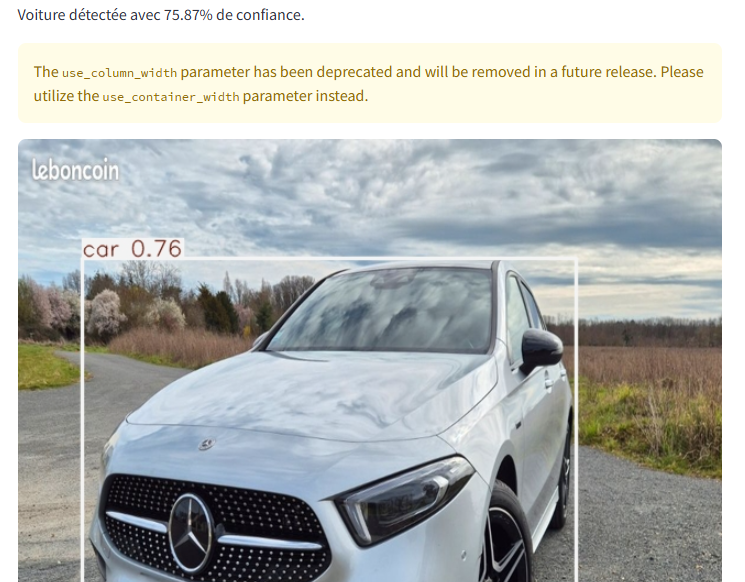
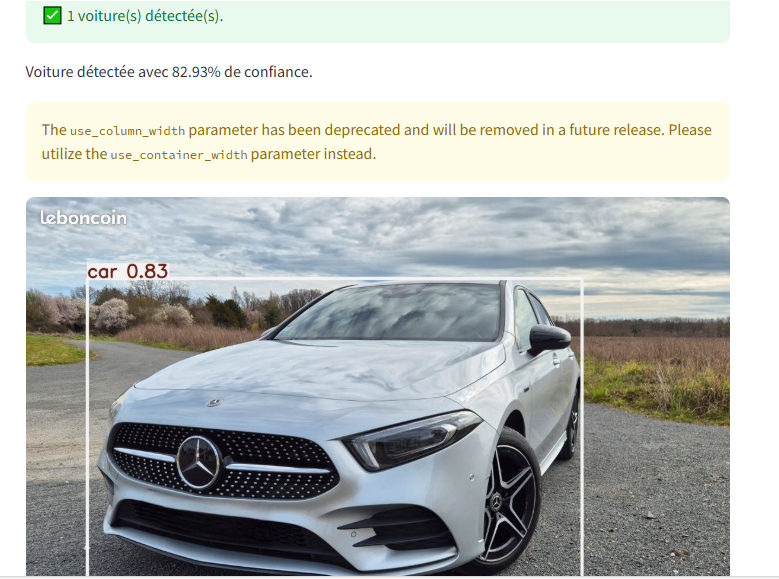


Image sans redimensionnement (score plus élevé, meilleure détection) :



D’un point de vue UX/UI, l’application est simple et fluide, mais une amélioration possible serait d’ajouter une visualisation dynamique du score de confiance sur l’image elle-même, en plus du tableau, pour aider l’utilisateur à interpréter visuellement la fiabilité de chaque détection. Une autre idée personnelle d'amélioration UX serait d'ajouter un mode "comparatif" qui permettrait de voir côte à côte l'image originale et l'image annotée, ou même d'afficher plusieurs résultats d’analyse en parallèle pour mieux comparer l’efficacité du modèle.

En résumé, ce projet démontre une application pratique de la détection d’objets avec un bon niveau de robustesse, tout en mettant en lumière les limites et pistes d’amélioration tant sur le plan technique que visuel.