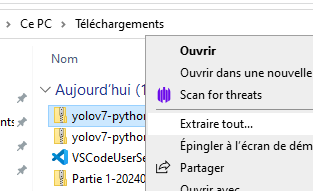
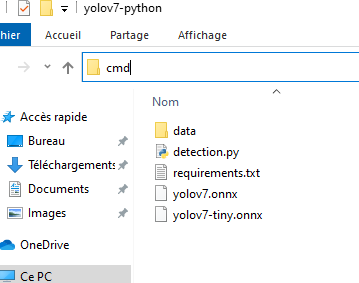
***Démo YOLO : (Regardez en bas du document)***

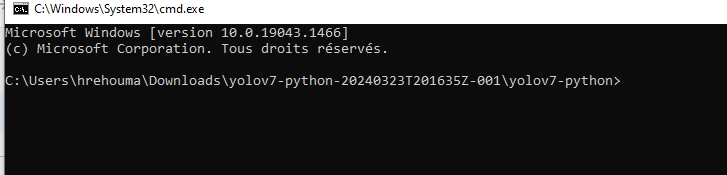
1. Téléchargez le fichier suivant : <https://drive.google.com/drive/folders/1eG7w9ZT7subxZQ9hwcvk3b85dCdL_R6S?usp=sharing>
2. Dézippez le fichier :



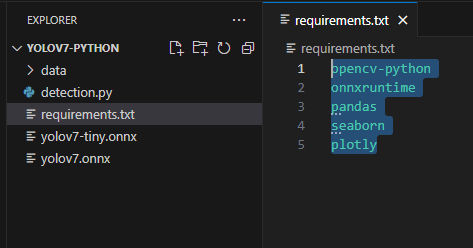
1. Allez dans le dossier extrait
2. Placez-vous à l’intérieur du dossier **yolov7-python**
3. Remplacez le chemin dans la barre de recherche en haut par la commande suivante : **cmd**

****

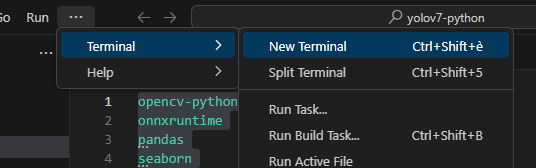
1. Cliquez sur entrée

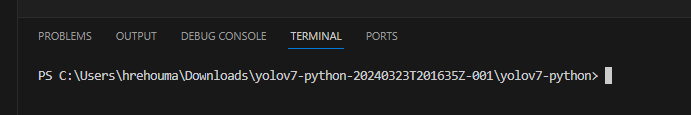


1. Pour afficher le contenu de votre répertoire. saisir la commande suivante : dir
2. écrire la commande suivante : **code** . **(attention !!! il y a un espace entre code et le point)**
3. Cette commande ouvrira le projet dans Vscode.
4. Ajoutez les dépendances suivantes dans requirements.txt

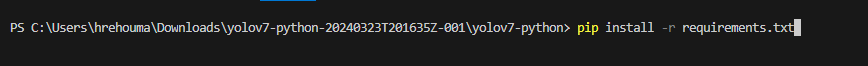
****

1. **Ouvrir le terminal**

****

****

1. Écrire la commande suivante : pip install -r requirements.txt

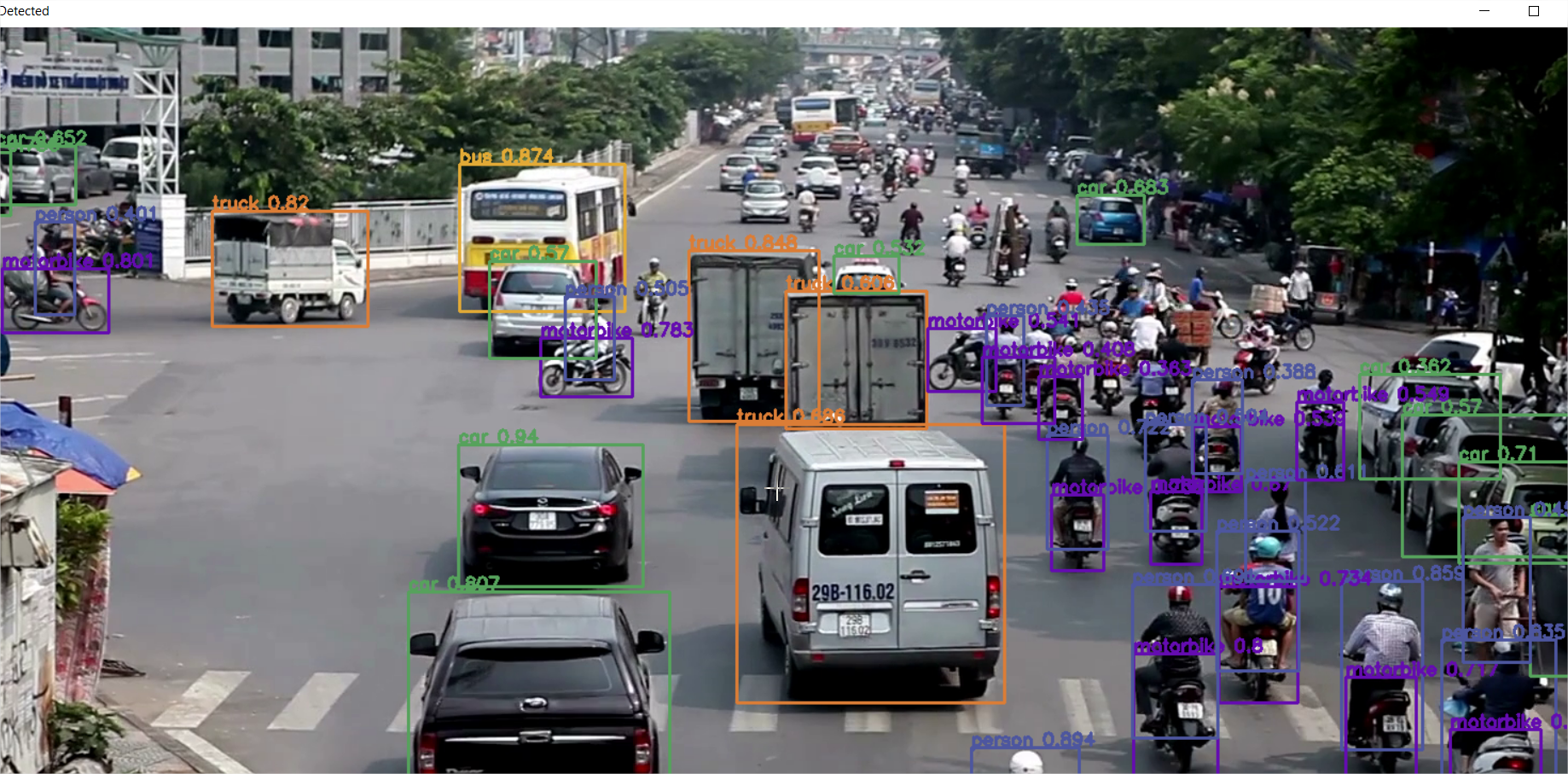




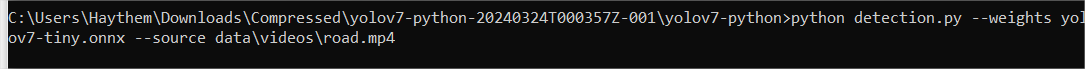
1. Écrire la commande suivante :



python detection.py --weights yolov7.onnx --source data\videos\road.mp4

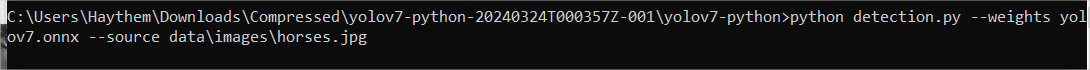


1. Cliquez sur Q pour quitter
2. Exécutez la commande suivante :



python detection.py --weights yolov7-tiny.onnx --source data\videos\road.mp4

1. Exécutez la commande suivante :



python detection.py --weights yolov7.onnx --source data\images\horses.jpg

1. Exécutez la commande suivante :

python detection.py --weights yolov7.onnx --source 0

1. YOLO 8

18.1. Téléchargez ce dossier : <https://drive.google.com/drive/folders/1-O2maCmNsMKwGejuyzOYix3bHOAfMFFB?usp=sharing>

18.2. pip install -r requirements.txt

18.3. python detection.py --model yolov8n.onnx --source 0

Résumé :

YOLO, pour "You Only Look Once", est un type d'algorithme de détection d'objets très populaire dans le domaine de la vision par ordinateur, principalement pour sa rapidité et son efficacité. Depuis la première version de YOLO, plusieurs itérations ont été développées, chacune visant à améliorer la précision, la vitesse ou la capacité de généralisation de l'algorithme.

YOLO7 et YOLO8 représentent deux versions successives de cette série d'algorithmes. Bien que les détails spécifiques puissent varier selon les implémentations et les mises à jour, voici quelques distinctions générales qui peuvent caractériser les évolutions entre deux versions consécutives comme YOLO7 et YOLO8 :

**1. Améliorations de la précision :** chaque nouvelle version de YOLO vise généralement à améliorer la précision de la détection des objets. Cela peut être réalisé par de meilleures architectures de réseau neuronal, des techniques d'augmentation de données plus avancées, ou des stratégies de post-traitement plus sophistiquées.

**2. Optimisation de la vitesse :** la vitesse de traitement est cruciale pour les applications de détection d'objets en temps réel. Les nouvelles versions cherchent à optimiser cela, soit en simplifiant l'architecture sans compromettre la précision, soit en utilisant des techniques de calcul plus efficaces.

**3. Robustesse et généralisation :** une meilleure capacité à généraliser à partir de divers ensembles de données et à fonctionner dans différents environnements ou conditions d'éclairage est un autre domaine d'amélioration. Cela peut inclure des modifications dans la manière dont le modèle est entraîné, comme l'introduction de nouveaux ensembles de données ou de méthodes d'entraînement.

**4. Fonctionnalités supplémentaires :** de nouvelles versions peuvent introduire des fonctionnalités supplémentaires, telles que la capacité à détecter des objets à différentes échelles plus efficacement, ou à mieux gérer les occlusions (lorsqu'un objet est partiellement caché par un autre).

**5. Compatibilité et intégration :** des efforts sont également faits pour rendre les modèles plus faciles à intégrer dans différentes plateformes ou pour améliorer leur compatibilité avec différents cadres de vision par ordinateur et bibliothèques d'apprentissage automatique.

Pour obtenir des informations spécifiques sur les différences entre YOLO7 et YOLO8, il serait idéal de consulter les documents officiels ou les articles de recherche publiés par les développeurs de ces algorithmes, car les détails techniques peuvent être assez spécifiques et dépendants des objectifs et des améliorations visées par chaque nouvelle version.