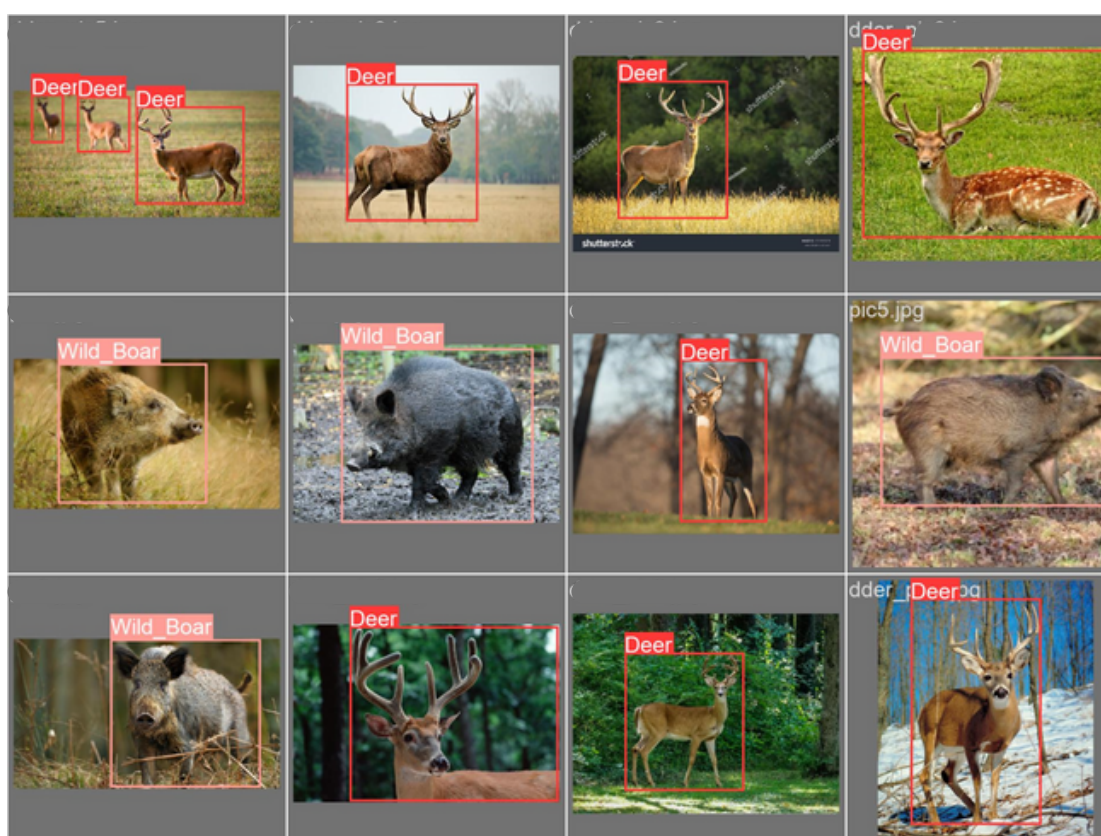


TEST TECHNIQUE

DÉTECTION ET CLASSIFICATION D'ANIMAUX SAUVAGES



SOMMAIRE

- 01** Présentation de l'API
- 02** Outils utilisés
- 03** Tutoriel d'utilisation

PRÉSENTATION DE L'API

Lien du dépôt Github contenant le code : <https://github.com/Armellectx/detection-wild-animals>

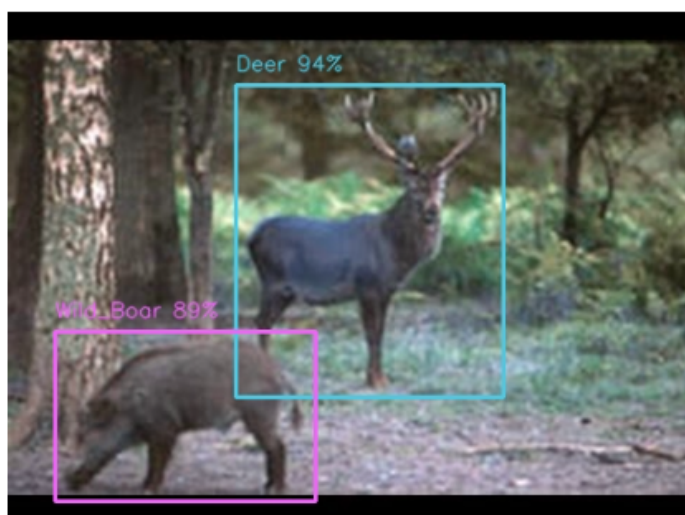
L'API proposée est accessible localement en executant un script Python. Un tutoriel est disponible dans la troisième section de ce rapport ainsi que sur le lien github fourni.

L'API permet de charger une image et de détecter la présence ou non de sangliers et de cerfs sur l'image. Chaque sanglier et cerf détecté est encadré par un rectangle de couleur, précisant le label de l'animal "Wild_Boar" ou "Deer", ainsi qu'un coefficient de certitude de la classification qui varie entre 0.5 et 1.

Les différentes prédictions effectuées sont sauvegardées au fur et à mesure dans un fichier "prédicitons.csv".



 hqdefault.jpg 26.2KB



Au total 2 animaux sauvages détectés sur cette image.

Nombre de sanglier(s): 1

Nombre de cerf(s): 1



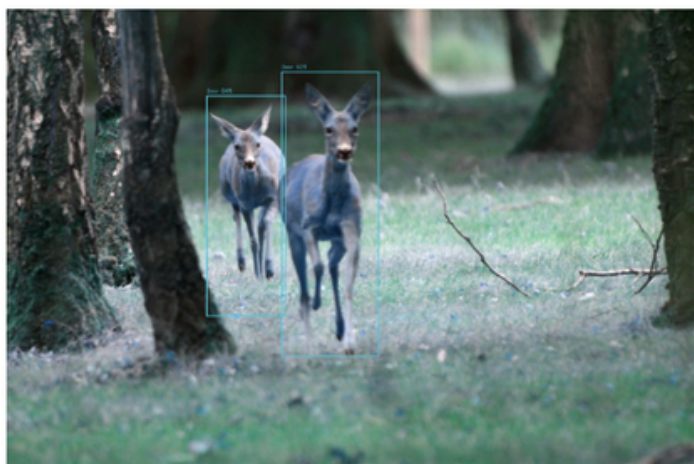
tnc_87536042.jpg 1.2MB



Pas de cerfs ni de sangliers détectés sur cette image



DSF_3817.jpg 459.2KB



Au total 2 animaux sauvages détectés sur cette image.

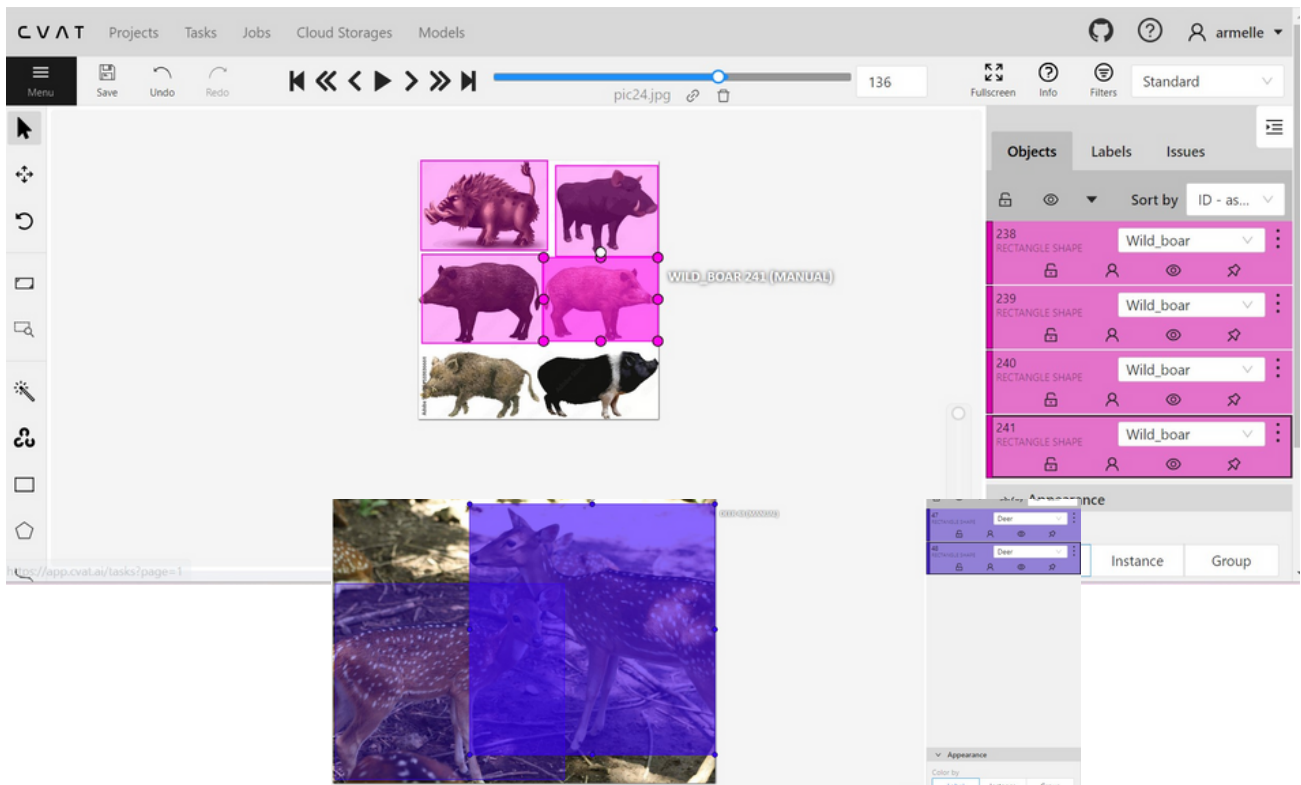
Nombre de cerf(s): 2

OUTILS ET DÉMARCHES

Choix et entraînement d'un CNN

Pour la détection d'animaux sauvages, j'ai fait le choix d'utiliser le modèle open source de détection d'objets YOLO (yolov8).

J'ai labélisé à la main une partie du jeu de données fourni grâce à l'outil de labélisation en ligne CVAT (Computer Vision Annotation Tool), illustré ci-dessous.



Déploiement de l'API

Pour l'interface utilisateur, j'ai utilisé streamlit, qui est une librairie à installer dans l'environnement virtuel Python de travail. C'est une interface agréable et facile à utiliser. Le déploiement de l'API sur le web est possible, mais payant ou limité en ressource.

Les temps de réponses semblent néanmoins supérieurs à ceux obtenus en explorant une solution avec Flask, qui permet une communication client-serveur plus efficace.

TUTORIEL D'UTILISATION

Etape 1

Rendez-vous sur gitbhub à l'adresse : <https://github.com/Armellectx/detection-wild-animals> et clonez le repository.

Etape 2

Dans un terminal, placez-vous dans le répertoire de l'application et installez les packages nécessaires avec la commande `pip install -r requirements.txt`.

Vous pouvez créer un environnement virtuel python à votre convenance.

Etape 3

Exécutez dans le terminal la commande: `streamlit run app.py`

L'application se chargera. Vous pouvez maintenant charger les images tests disponibles dans le répertoire `images/`, ou bien vos propres images à **enregistrer au préalable dans le répertoire `images/`**.