



Développer une preuve de concept YOLOv9 dans la classification d'images



Lien vers le Github : https://github.com/Bastien441237/P7_OpenClassroomsProject.git

Sommaire

1. Contexte du projet	3
2. Sources	4
3. Choix de l'algorithme récent	5-6
4. Exploration des données	7-8
5. Nettoyage des données	9-11
6. VGG16 : modèle baseline	12
7. Création des fichiers labels	13
8. Modélisation avec YOL0v9	14
9. Synthèse des résultats	15-18
10. Choix du modèle	19
11. Mise en place du Dashboard	20-22
12. Démonstration du Dashboard	23
13. Conclusion	24

Contexte du projet

Problématique : Réaliser une preuve de concept sur le modèle YOLOv9, modèle de détection d'objet, dans le cadre d'une classification d'images.



Objectif : Essayer de prouver que le modèle YOLOv9 peut être utilisé dans le cadre d'une classification de races de chiens et montrer qu'il peut être meilleur qu'un réseau de neurones classique.



Missions :

- Explorer les données du Stanford Dogs Dataset.
- Nettoyer les données.
- Réaliser une baseline pour comparer aux résultats de YOLOv9.
- Entraîner le modèle YOLOv9 sur les données personnalisées.
- Réaliser un dashboard à l'aide de Streamlit.



Résultat attendu :

- Un plan de travail prévisionnel.
- Un notebook contenant l'analyse des données et la modélisation.
- Une note méthodologique présentant la preuve de concept.
- Un dashboard Streamlit déployé sur le cloud.

Sources

Stanford Dogs Dataset

- Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao and Li Fei-Fei. Novel dataset for Fine-Grained Image Categorization. First Workshop on Fine-Grained Visual Categorization (FGVC), IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011.
- J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li and L. Fei-Fei, ImageNet: A Large-Scale Hierarchical Image Database. IEEE Computer Vision and Pattern Recognition (CVPR), 2009.

Yolov9

- <https://arxiv.org/abs/2402>
- <https://docs.ultralytics.com/fr/models/yolov9/#citations-and-acknowledgements>
- <https://blent.ai/blog/a/detection-images-yolo-tensorflow>
- <https://github.com/WongKinYiu/yolov9>
- <https://github.com/AarohiSingla/YOL0v9>
- <https://www.youtube.com/@CodeWithAarohi>

Choix de l'algorithme récent .1/2

Principes mathématiques

Principe de l'entonnoir d'information

$$I(X, X) \geq I(X, f_\theta(X)) \geq I(X, g_\phi(f_\theta(X)))$$

Perte progressive d'information lors de la transformation des données dans les réseaux de neurones profonds

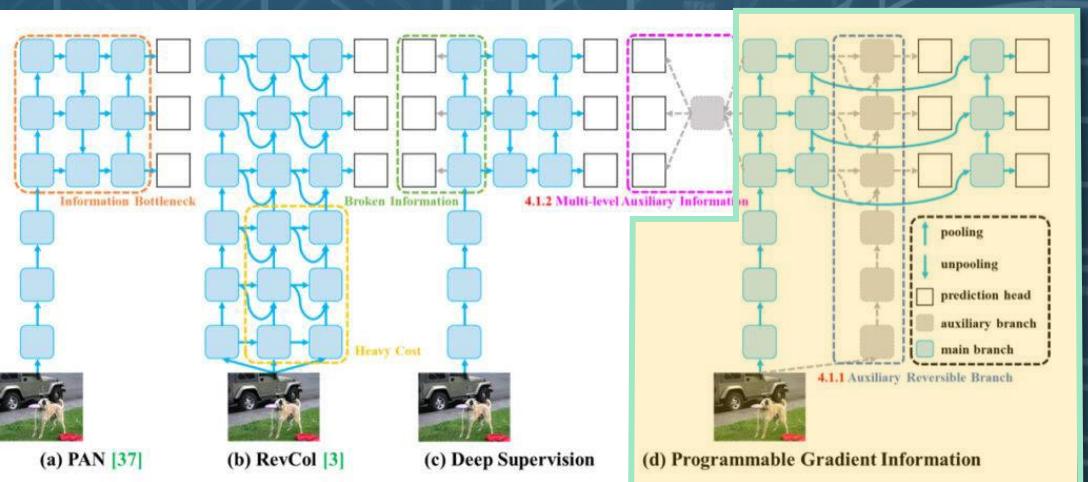
Résolutions possibles envisagées :

- Augmentation de la taille du modèle
- Utilisation de fonctions réversibles

Fonction réversible

$$X = v_\zeta(r_\psi(X))$$

Données X transformées par une fonction réversible sans perte d'information



Le principe de l'entonnoir d'information et la mise en place de fonctions de transformation réversible sont des points essentiels dans la convergence et l'efficacité de réseaux de neurones profonds.

Choix de l'algorithme récent .2/2

YOLOv9

Distinctions du modèle YOLOv9 par rapport aux Modèles Classiques :

1. Vitesse de Traitement

En effectuant toutes les étapes de la détection en un seul passage, YOLOv9 est capable de détecter les objets en temps réel, ce qui le rend idéal pour les applications nécessitant une réponse rapide.

2. Efficacité et Précision

L'utilisation de techniques telles que PGI et GELAN permet d'optimiser les performances du modèle tout en réduisant la consommation de ressources.

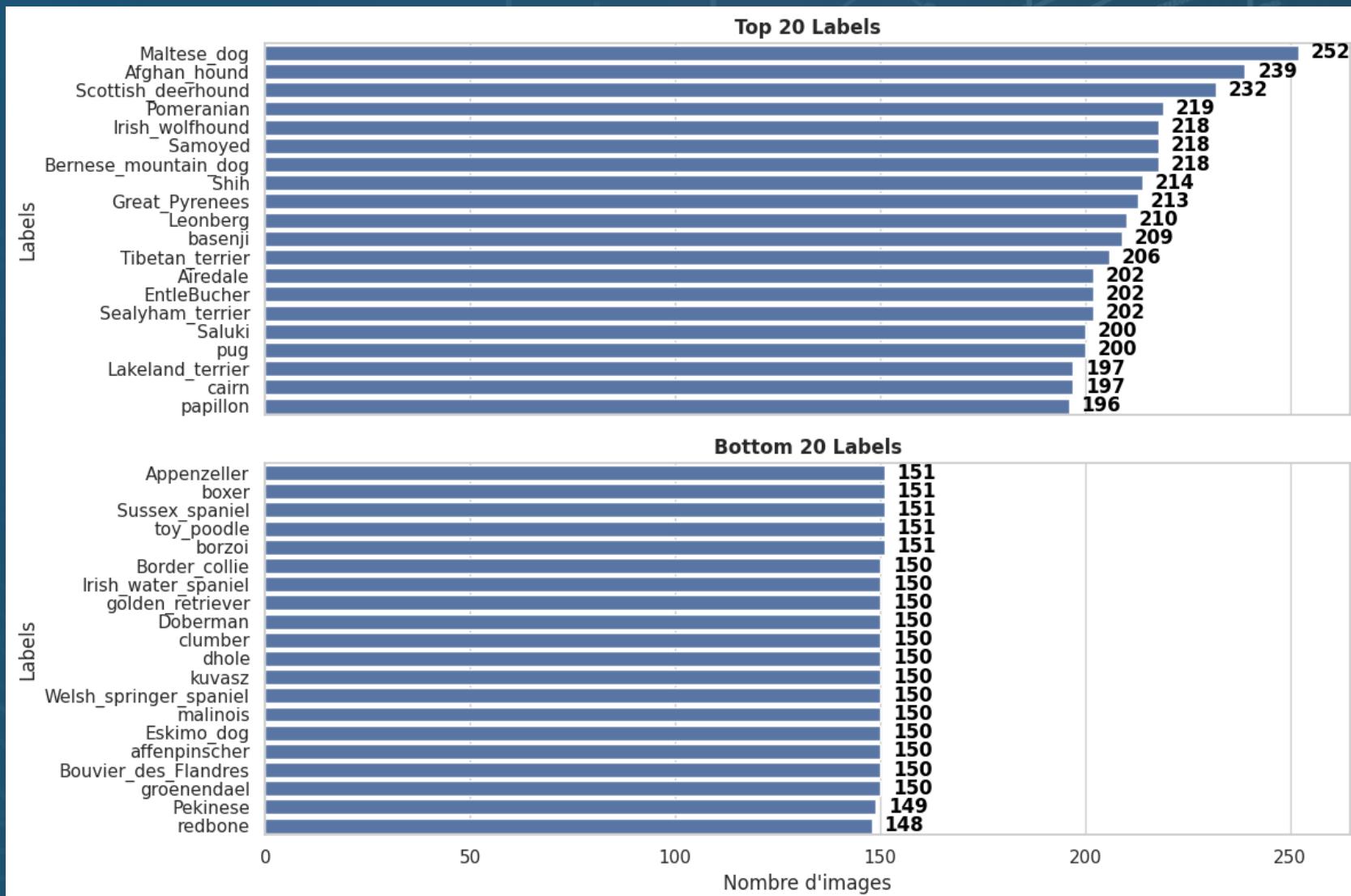
3. Adaptabilité

Sa capacité à fonctionner efficacement sur différents types de données et dans des environnements variés en fait un choix polyvalent pour de nombreuses applications de vision par ordinateur (ex : conduite autonome, surveillance de la sécurité, imagerie médicale, etc.)

Exploration des données .1/2

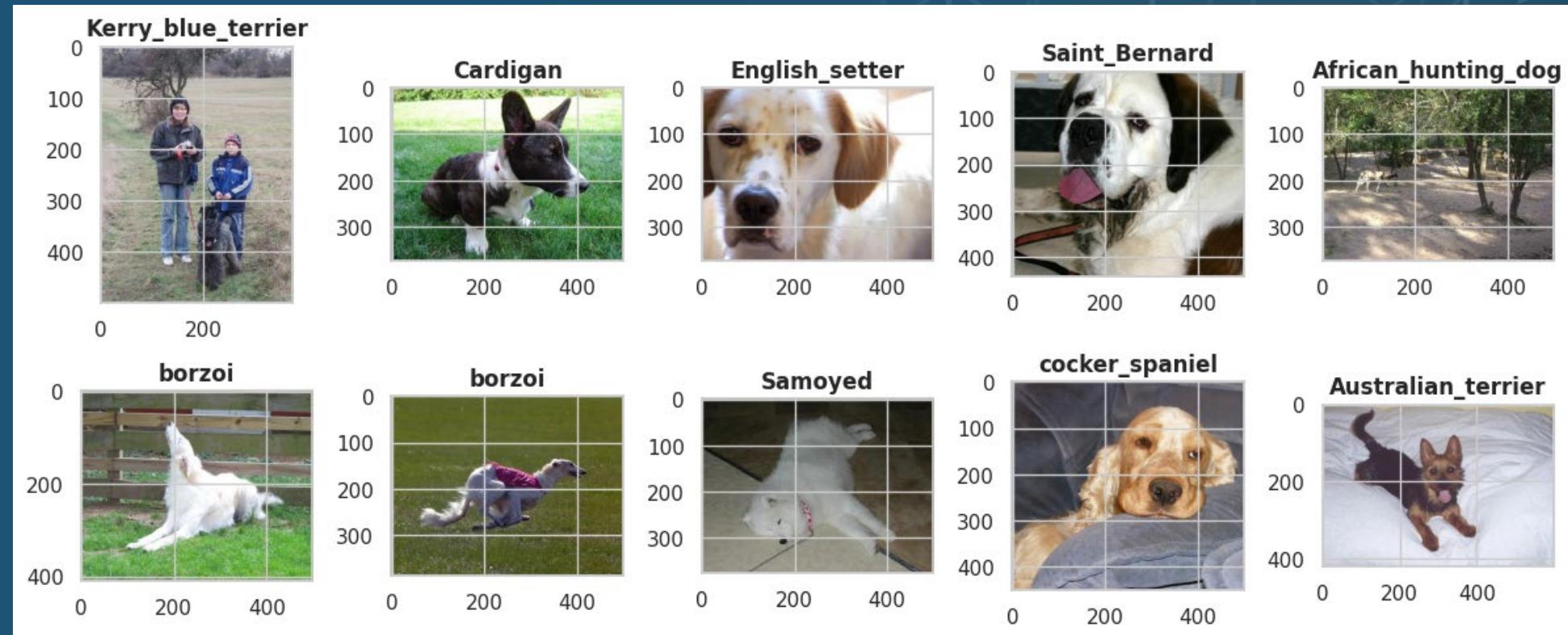
Images disponibles

Le Stanford Dogs DataSet contient 120 races de chiens et plus de 20 580 images, nous avons à notre disposition 148 à 252 images par race.



Exploration des données .2/2

Exemple d'images composants le DataSet



Sur cet exemple d'images, nous pouvons voir que celles-ci ne sont pas de la même dimension.
Certaines images comportent même des objets ou la présence d'humains voire d'autres chiens.
Il va falloir réaliser un prétraitement de ces images.

Nettoyage des données .1/3

Echantillonnage du dataset

Une étape préliminaire de nettoyage a été entreprise afin d'assurer la qualité des données : 15 races de chiens similaires ont été sélectionnées de manière stratégique afin :

- de garantir un échantillon représentatif des performances que nous pourrions atteindre sur l'ensemble du Stanford Dogs Dataset,
- tout en réduisant le temps de calcul des modèles.



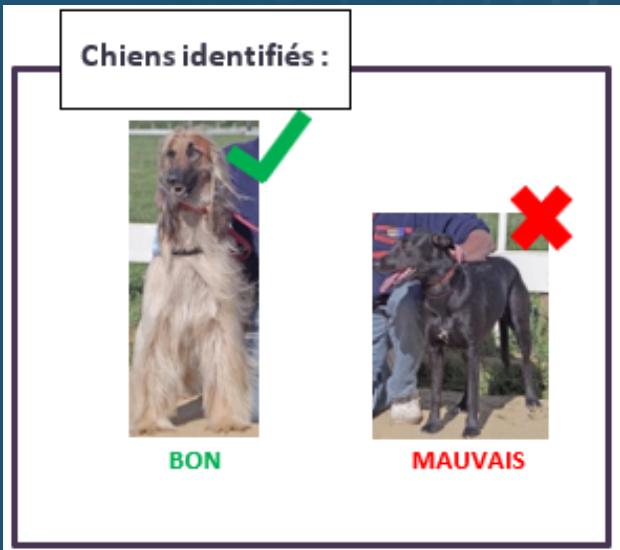
```
Found 2030 images belonging to 15 classes.
{0: 'Bernese_mountain_dog',
 1: 'boxer',
 2: 'briard',
 3: 'Brittany_spaniel',
 4: 'bull_mastiff',
 5: 'Doberman',
 6: 'EntleBucher',
 7: 'French_bulldog',
 8: 'Gordon_setter',
 9: 'Greater_Swiss_Mountain_dog',
 10: 'Irish_setter',
 11: 'Rottweiler',
 12: 'Samoyed',
 13: 'Tibetan_terrier',
 14: 'vizsla'}
```

Nettoyage des données .2/3

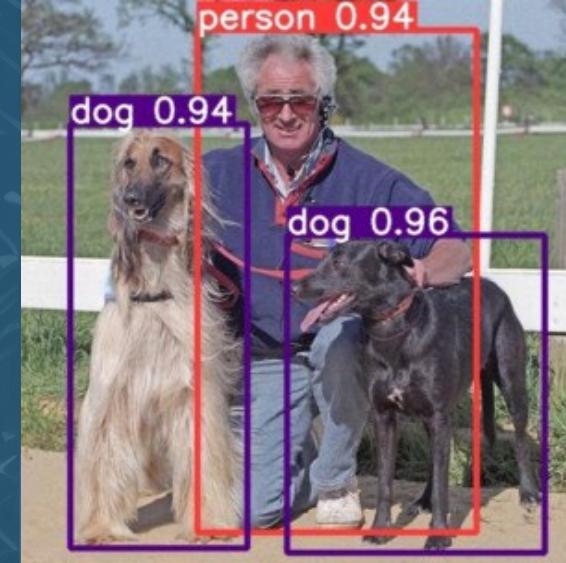
Détection des chiens sur les images sélectionnées

1

l'algorithme YOLOv9 a été utilisé pour détecter les chiens sur les images, permettant ainsi de ne conserver que les parties pertinentes contenant des chiens.



2



Pour assurer la précision des données, une vérification manuelle a été effectuée pour confirmer que les images extraites correspondaient bien à la race indiquée par l'étiquette.

Nettoyage des données .3/3

Détourage des images avec Bria AI (RMBG 1.4)

Enfin, afin de réduire au maximum le bruit dans les images, nous avons opté pour l'utilisation du modèle de Bria AI (RMBG 1.4), qui offre la possibilité de détourer l'image.



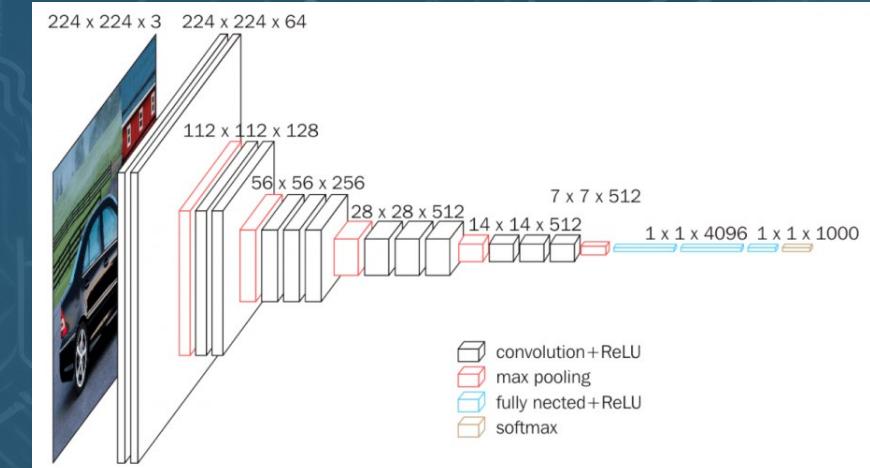
A noter : afin de garantir la conservation de suffisamment d'informations, nous testerons l'entraînement du modèle YOLOv9 avec ET sans le détourage de l'image.

VGG16 : modèle baseline

La baseline choisie est VGG16, un réseau de neurones profond utilisé principalement pour la classification et entraîné sur Imagenet (dont notre base de données fait partie).

Cela nous permettra de comparer les résultats obtenus via VGG16 aux résultats obtenus via YOLOv9.

En effet, notre approche diffère ici de l'utilisation traditionnelle de YOLOv9 (qui se limite à la détection d'objets) : nous souhaitons explorer la possibilité d'entraîner YOLOv9 à reconnaître les différentes races de chiens, similairement à un modèle de classification.



Architecture du modèle VGG16

A noter : Nous avons entraîné ce modèle pré-entraîné sur les images brutes, c'est-à-dire non nettoyées. Mais nous avons utilisé les couches d'input de VGG16 pour le preprocessing de celles-ci.

Création des fichiers labels

L'organisation initiale du jeu de données ne convient pas à l'entraînement de YOLOv9.

Il est donc nécessaire de réorganiser l'arborescence des dossiers :

- 1/ en encodant les labels dans un fichier .txt
- 2/ en fournissant le fichier .yaml qui permet de « décoder » les prédictions

Ancienne structure :

Dossier du DataSet > Train/Validation/Test > Labels > nom_de_l'image.png

Nouvelle structure :

Dossier du DataSet > Train/Validation/Test > images/labels > images.jpg/nom_de_l'image.txt

Modélisation avec YOLOv9

Le but est de définir si l'on obtient plus de performance avec des images détournées et si le modèle est améliorable en l'optimisant

PREMIERE PHASE =

Entraînement sur :

- les images non détournées
- sur 70 epochs

DEUXIEME PHASE =

Entraînement sur :

- les images détournées
- sur 70 epochs

TROISIEME PHASE =
Optimisation du modèle

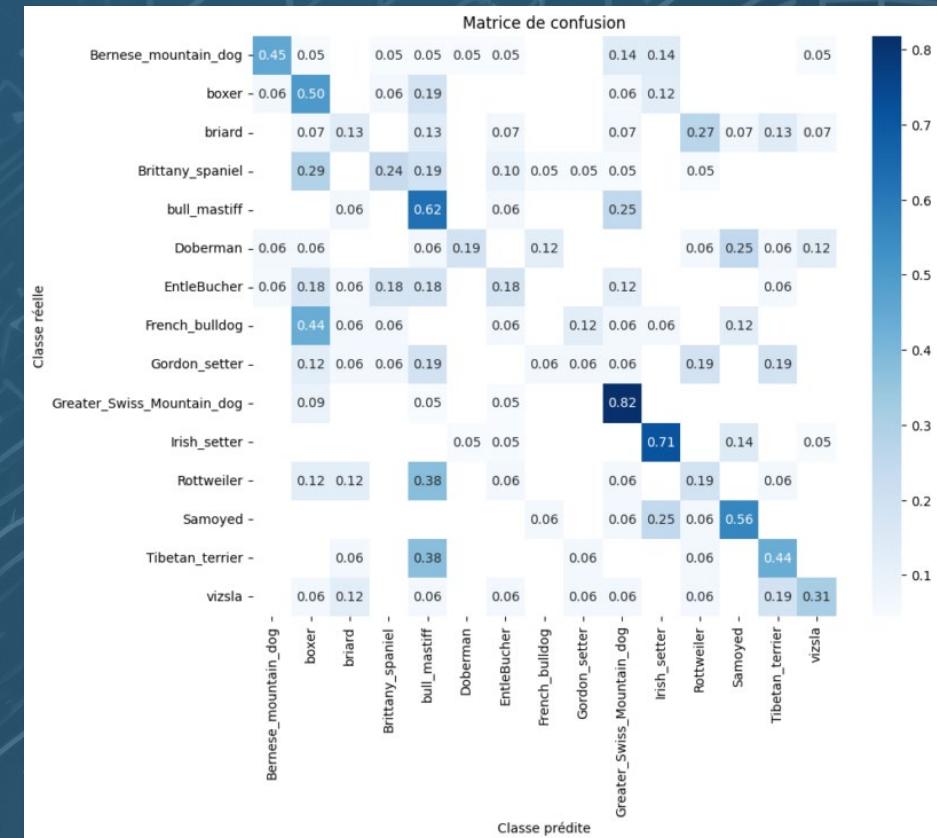
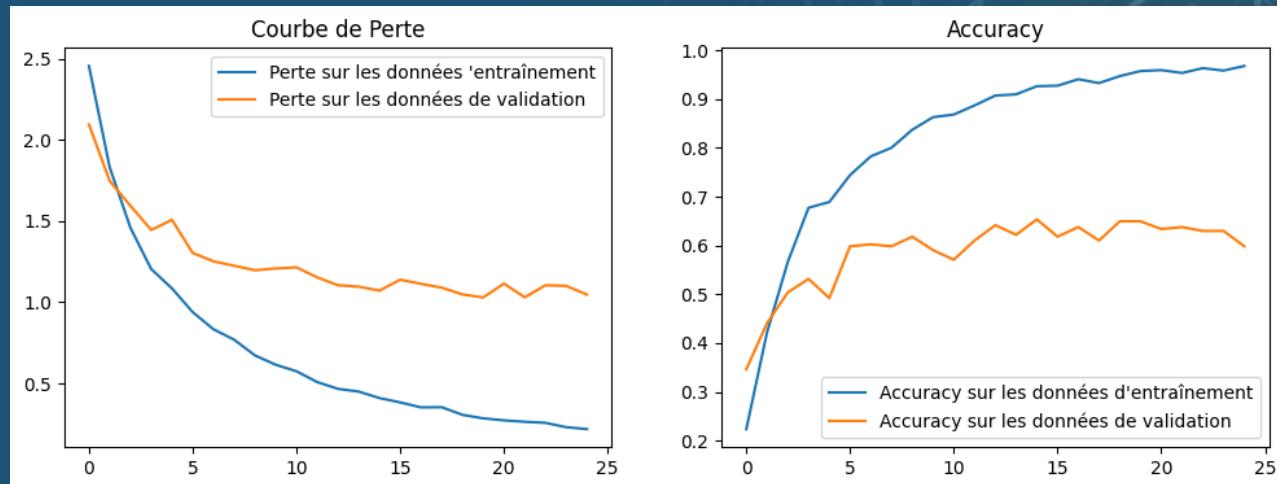


Synthèse des résultats .1/4

Résultat du modèle VGG16

Lors de son entraînement, VGG16 a tendance à présenter des signes de surapprentissage, même avec l'utilisation d'un EarlyStopping pour limiter le nombre d'epochs. Cependant, cette tendance est mieux comprise lorsque nous examinons les scores de la matrice de confusion. Comme prévu avec l'échantillonnage des 15 races de chiens, le modèle éprouve des difficultés à distinguer les races similaires.

Cette baseline nous fournit un premier score d'accuracy de 0.65 à partir duquel nous allons pouvoir évaluer les performances de YOLOv9, ainsi que l'impact de notre processus de nettoyage préalable.

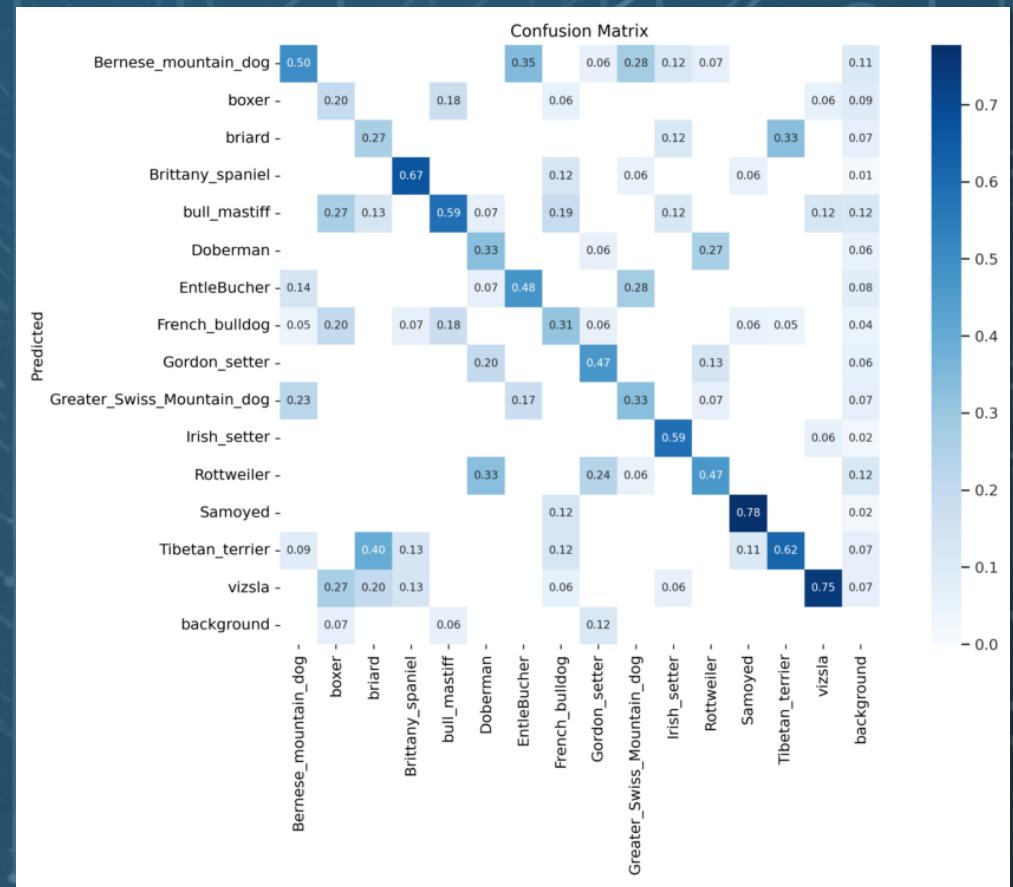
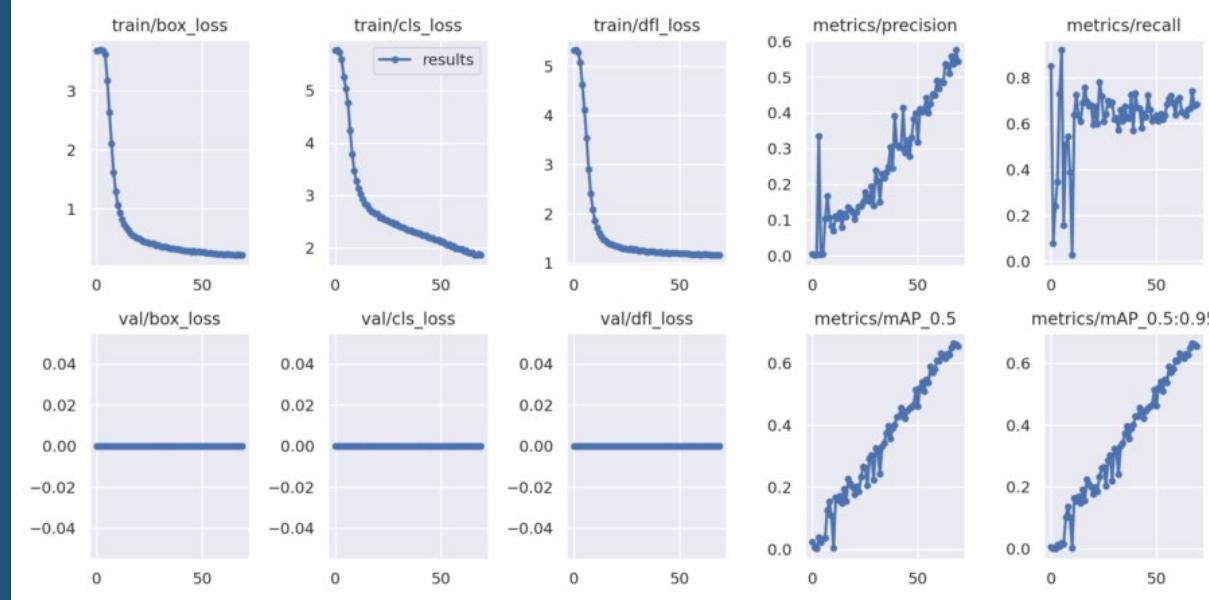


Synthèse des résultats .2/4

Résultat du modèle YOL0v9 sur images non détournées

Le modèle YOL0v9 a été entraîné sur 70 epochs en utilisant les images non détournées, aboutissant à une accuracy maximale de 0.63. Cette performance est inférieure à celle du modèle VGG16.

Cependant, une analyse plus approfondie de la matrice de confusion révèle que le modèle commence à discerner les races de chiens, bien qu'il éprouve encore des difficultés avec les races similaires.

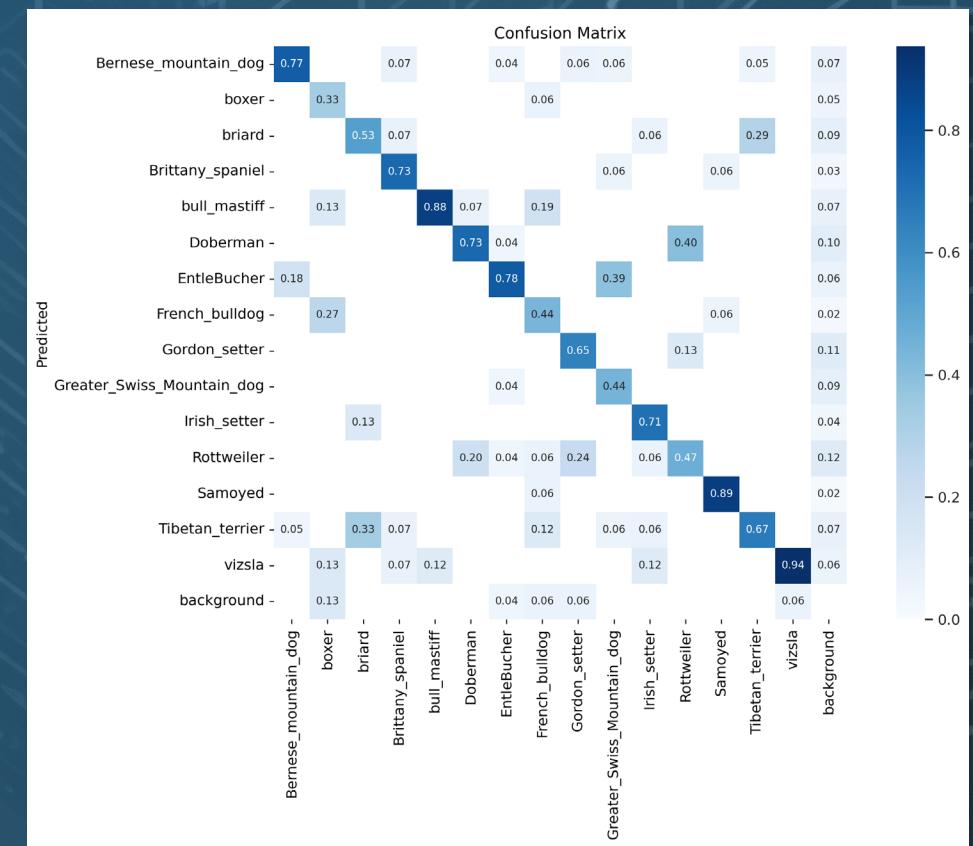
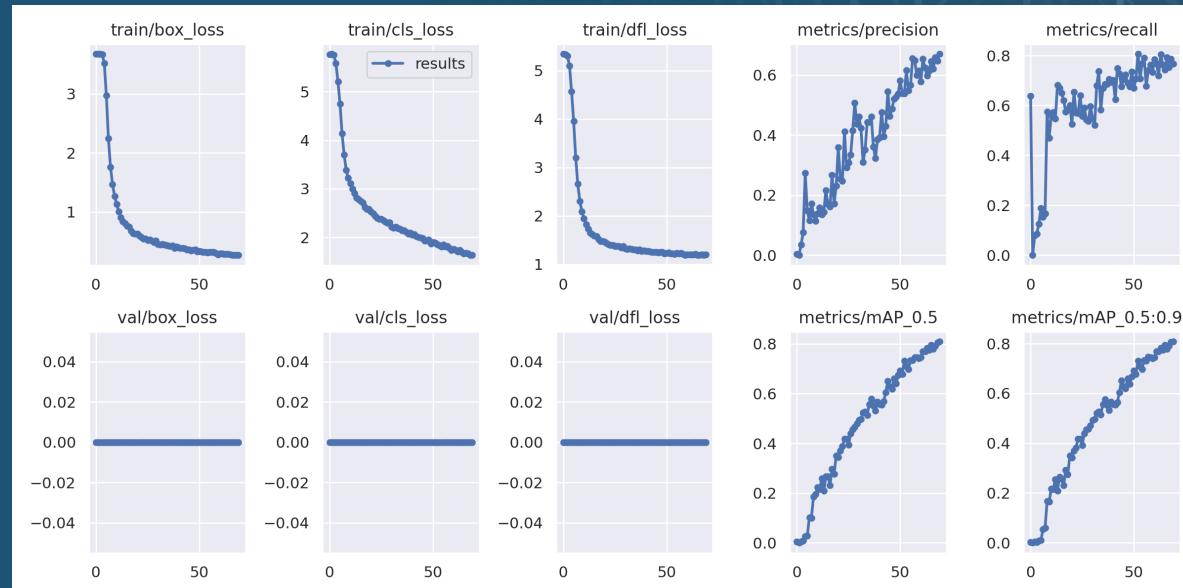


Synthèse des résultats .3/4

Résultat du modèle YOL0v9 sur images détournées

Ensuite, nous avons entraîné le modèle YOL0v9 sur les images détournées, ce qui a donné des résultats améliorés sur le même nombre d'epochs (70), avec un score d'accuracy de 0.75.

Cependant, il était évident que la précision pouvait être davantage améliorée en augmentant le nombre d'epochs, tout en surveillant le risque de surapprentissage.

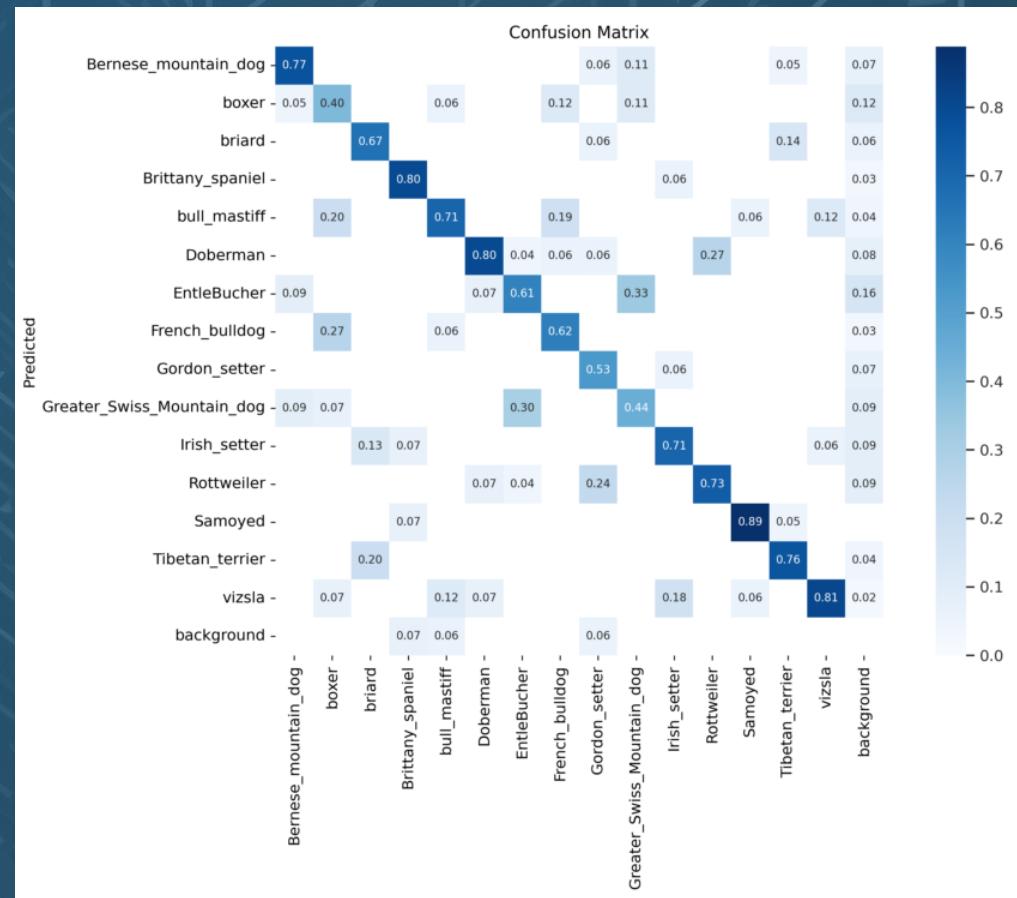
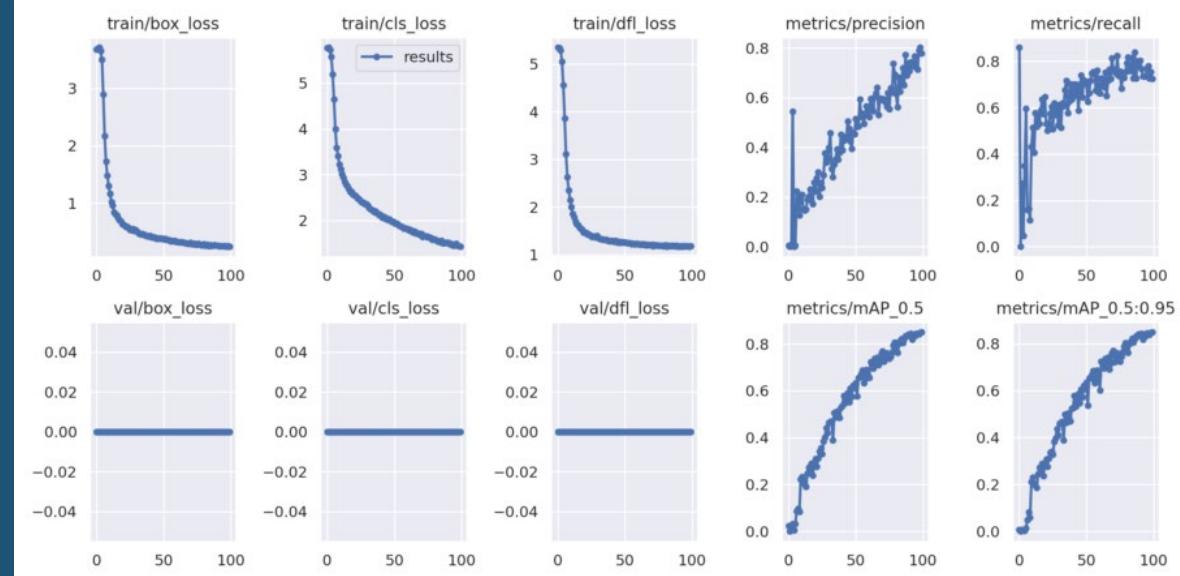


Synthèse des résultats .4/4

Résultat du modèle YOL0v9 sur images détournées avec plus d'epochs

Nous avons augmenté le nombre d'epochs à 100, ce qui nous a permis d'atteindre une accuracy d'environ 0.73.

Cependant, l'analyse de la matrice de confusion révèle une amélioration dans la reconnaissance des races de chiens similaires, ainsi qu'une amélioration globale des classifications.



Choix du modèle

En conclusion, bien que notre approche de modélisation ait donné des résultats prometteurs, il reste encore des défis à relever pour améliorer sa performance et son interprétabilité. En explorant ces pistes d'amélioration, nous pourrons continuer à affiner notre modèle et à le rendre plus efficace pour la reconnaissance des races de chiens dans diverses applications.

	VGG16 Modèle baseline	YOLOv9 Sur images non détournées	YOLOv9 Sur images détournées	YOLOv9 Sur images détournées avec plus d'epochs
Nombre d'epochs	25	70	70	100
Accuracy (Test set)	0.65	0.63	0.75	0.73

Mise en place du Dashboard .1/3

Contenu du Dashboard

Le Dashboard permet à l'utilisateur de comprendre le processus de modélisation avec YOLOv9 de manière interactive.

C'est pour cela que l'on y retrouvera 5 pages différentes vers lesquelles l'utilisateur peut naviguer grâce au sommaire mis à disposition :

Un rappel du contexte du projet

Un explication du contenu du dataset avec la possibilité d'explorer les données

L'explication du nettoyage de ces données avec visualisation étape par étape des données nettoyées

Une synthèse des résultats obtenus au fur et à mesure de l'entraînement du modèle avec visualisation et possibilité de comparer de manière interactive les prédictions de YOLOv9 par rapport au modèle baseline

Une API pour tester le modèle, afin de prédire la race de chien à partir de la banque d'images mise à disposition (hors dataset de base) ou en chargeant une image personnelle

Sommaire

Aller à la page :

- Contexte du projet
- Analyse exploratoire des données
- Nettoyage des données
- Choix du modèle
- Prédiction du modèle

Mise en place du Dashboard .2/3

Les différentes fonctionnalités incluses

Afin de rendre le dasboard interactif pour l'utilisateur les fonctionnalités suivantes ont été mises en place :

Sélectionner les labels :

Ibizan_hound x Staffordshire_bu... x

Nombre d'images par labels :

3

1 10

La sélection d'un ou plusieurs labels et la possibilité d'afficher entre 1 et 10 images par label

Choix des paramètres de prédition test :

Choisir une race de chien :

Samoyed

Nombre d'images par race de chiens :

1

1 2

Prédir

Le choix d'une race de chien et d'un nombre d'images (entre 1 et 2, issues des données de test), afin de confronter les prédictions du modèle baseline et du modèle YOLOv9

Choix d'une image dans la banque d'image :

Sélectionner une image :

Tibetan_terrier.jpg



Tibetan_terrier.jpg

Veuillez charger votre image ci-dessous :

Charger l'image du chien :

Drag and drop file here Limit 200MB per file • PNG, JPEG, JPG

Browse files

Prédir avec YOLOV9

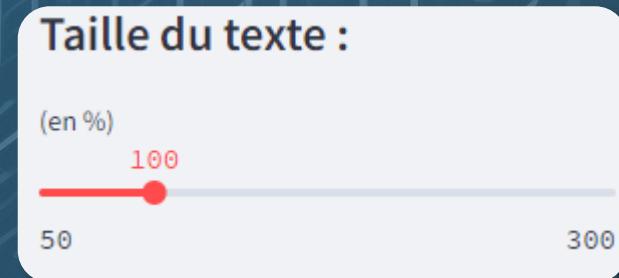
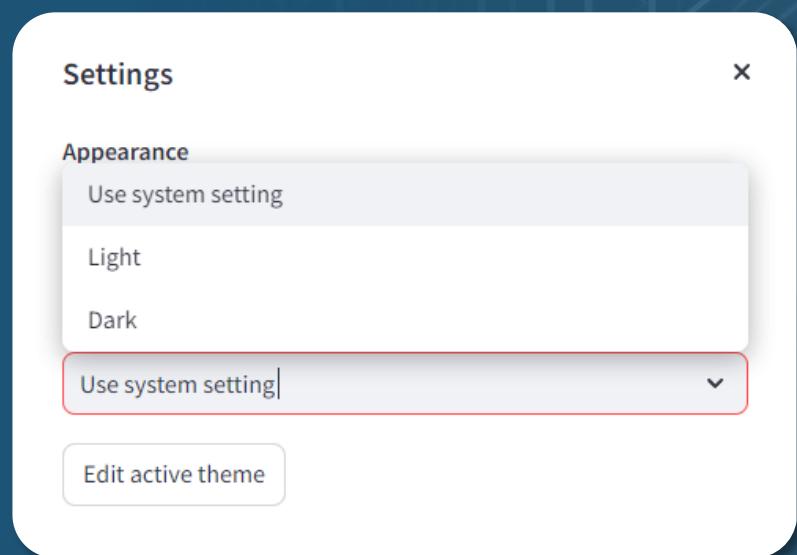
La possibilité de choisir une image dans une banque de données de test spécialement conçu, ou bien d'importer une image afin de pouvoir réaliser des prédictions avec YOLOv9

Mise en place du Dashboard .3/3

Respect des critères d'accessibilité du WCAG

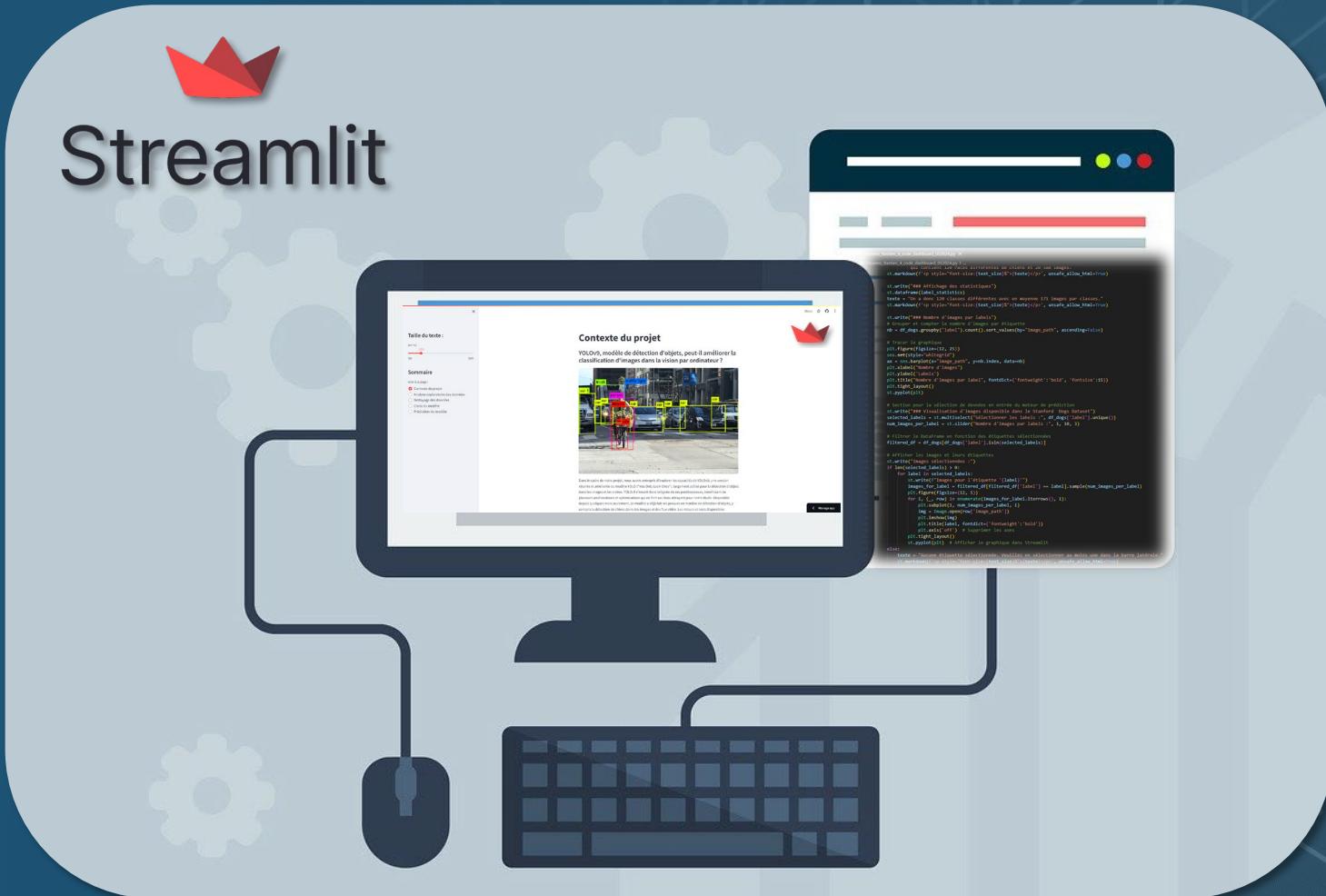
Nous avons également pris en compte le besoin des personnes en situation de handicap en couvrant des critères d'accessibilité du WCAG. Notamment sur les critères suivant :

- Contenu non textuel
- Utilisation de la couleur
- Contraste (possibilité d'avoir un « dark mode » et un « light mode »)
- Redimensionnement du texte
- Titre de page



Démonstration du Dashboard

Lien vers le dashboard : <https://dashboard-yolov9.streamlit.app/>



Conclusion

Défis et Perspectives

Malgré les résultats prometteurs, des défis persistent dans notre approche de modélisation pour la classification des races de chiens.

1. Limites de l'ensemble de données : La taille et la diversité limitée peuvent entraver la généralisation du modèle.
2. Interprétabilité du modèle : Comprendre les décisions du modèle peut être complexe.
3. Absence de bibliothèques dédiées : Tout le code étant en open source sur le Github, l'implémentation dans des projets personnalisés peut s'avérer complexe.