

NOTE METHODOLOGIQUE : PREUVE DE CONCEPT MODELE YOLOV9

Projet 7 OPENCLASSROOMS – Machine Learning Engineer

Bastien Moreno

TABLE DES MATIERES

Dataset retenu	3
Les concepts de l'algorithme récent.....	4
La modelisation.....	6
Nettoyage des données.....	6
Mise en place d'une baseline	7
Utilisation du modèle YOLOv9	7
Optimisation du modèle.....	8
Synthèse des résultats	9
Modèle VGG16.....	9
Modèle YOLOv9 sur les images non détournées.....	10
Modèle YOLOv9 sur les images détournées.....	11
Les limites et les améliorations possibles	12

DATASET RETENU

Le Stanford Dogs Dataset se révèle être une ressource cruciale pour la classification des races de chiens, abritant 120 races et plus de 20 580 images, avec à notre disposition de 148 à 252 images par race de chiens. Dans le cadre de notre étude, nous avons choisi de restreindre notre focus à seulement 15 races de chiens. Cette sélection a été motivée par la volonté de réduire le temps de calcul tout en conservant la représentativité de l'échantillon. Notre objectif principal est d'évaluer les performances des modèles sur des races similaires où des difficultés de classification sont souvent rencontrées.

Statistiques descriptives du DataSet :

Nombre d'images	
count	120.000000
mean	171.500000
std	23.220898
min	148.000000
25%	152.750000
50%	159.500000
75%	186.250000
max	252.000000
Total	20580.000000

Le Stanford Dogs Dataset, bien que riche en images de races de chiens, peut également présenter des éléments indésirables tels que la présence d'autres animaux, objets ou humains dans les images. Dans le cadre de notre étude, une étape cruciale consistera à nettoyer ces images afin de ne conserver que celles qui mettent en valeur les chiens que nous souhaitons classifier. Cette démarche garantira la qualité de notre ensemble de données en éliminant les distractions potentielles et en se concentrant uniquement sur les éléments pertinents pour notre tâche de classification.

Références :

Primary:

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.

Secondary:
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.

La détection d'objets en vision par ordinateur est une tâche cruciale qui implique l'identification et la localisation précise d'objets dans une image ou une séquence vidéo. Parmi les différents modèles de détection d'objets, YOLOv9 se distingue par son efficacité et sa précision remarquables, ainsi que par sa capacité à fonctionner en temps réel. Son fonctionnement repose sur plusieurs principes clés qui le différencient des modèles de réseau neuronal classiques.

Principes de Fonctionnement de YOLOv9 :

1. **Approche en Un Seul Passage** : YOLOv9 adopte une approche en un seul passage à travers le réseau neuronal pour détecter les objets dans une image. Contrairement aux modèles traditionnels qui nécessitent plusieurs itérations, YOLOv9 effectue toutes les étapes de la détection en un seul passage, ce qui lui confère une vitesse de traitement exceptionnelle.
2. **Division en Grille** : L'image d'entrée est divisée en une grille de cellules, et chaque cellule est responsable de prédire les boîtes englobantes et les probabilités de classe des objets dont les centres se trouvent à l'intérieur de cette cellule. Cette approche permet une localisation précise des objets dans l'image.
3. **Utilisation de Réseaux de Neurones Convolutifs (CNN)** : YOLOv9 utilise des réseaux de neurones pour extraire des caractéristiques pertinentes de l'image et effectuer la détection d'objets. Ces réseaux sont spécialement conçus pour la vision par ordinateur et sont capables de reconnaître des motifs complexes dans les données visuelles.
4. **Programmable Gradient Information (PGI)** : Une innovation clé de YOLOv9 est l'introduction de l'Information de Gradient Programmable (PGI), qui permet de préserver les caractéristiques importantes tout au long du réseau. Cela garantit que les informations cruciales ne sont pas perdues pendant le processus de détection.

En effet, il est important de comprendre le principe d'entonnoir d'information où la transformation des données fait perdre progressivement de l'information dans les réseaux de neurones classiques.

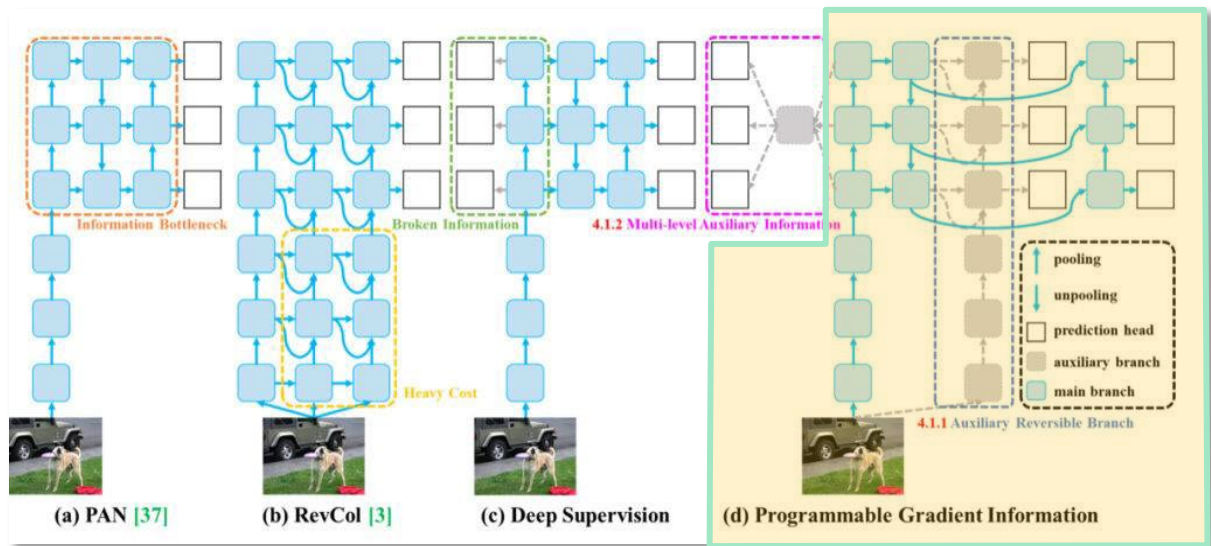
$$I(X, X) \geq I(X, f_{\theta}(X)) \geq I(X, g_{\phi}(f_{\theta}(X)))$$

Ici, les fonctions (f) et (g) représentent les fonctions de transformations successives dans un réseau de neurones classique. On peut voir que I (l'information) peut diminuer au fur et à mesure des transformations.

La solution pourrait être l'augmentation de la taille du modèle, mais ici on préfère utiliser une fonction réversible afin de transformer les données sans perte d'information.

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

On va appliquer la fonction de transformation réversible (r) et celle-ci est dite réversible si elle possède une fonction de transformation inverse (v), les paramètres étant placés à l'extérieur.



5. **Réseau d'Agrégation de Couches Efficace Généralisé (GELAN)** : YOLOv9 utilise également le Réseau d'Agrégation de Couches Efficace Généralisé (GELAN) pour optimiser l'utilisation des paramètres et l'efficacité de calcul dans les réseaux profonds. Cette technique permet une architecture légère tout en maintenant des performances élevées.

Distinctions par rapport aux Modèles Classiques :

1. **Vitesse de Traitement** : La principale distinction de YOLOv9 par rapport aux modèles classiques réside dans sa 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** : Malgré sa rapidité, YOLOv9 parvient à maintenir une efficacité et une précision élevées dans la détection d'objets. 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** : YOLOv9 est également très adaptable et peut être utilisé dans une variété d'applications, de la conduite autonome à la surveillance de la sécurité en passant par l'imagerie médicale. 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.

En résumé, YOLOv9 se distingue par son approche en un seul passage, son utilisation efficace des ressources et sa capacité à maintenir des performances élevées en temps réel. Ces caractéristiques en font un choix convaincant pour les tâches de détection d'objets dans une large gamme d'applications.

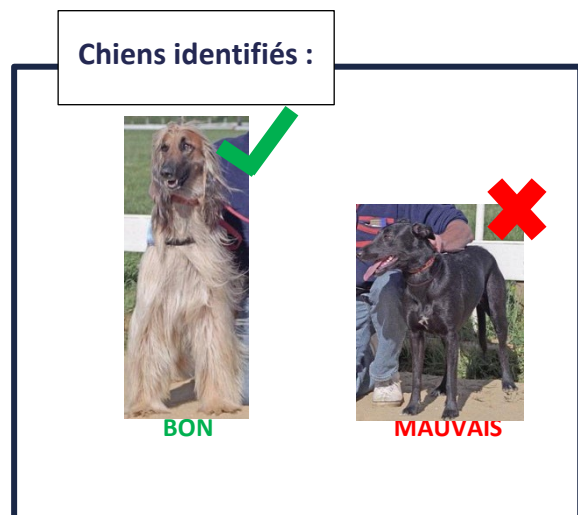
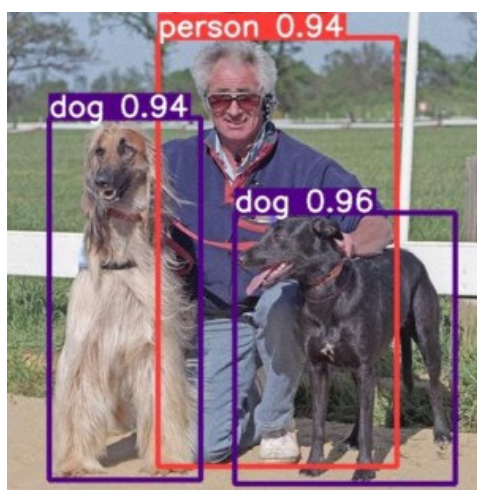
LA MODELISATION

La méthodologie de modélisation déployée dans ce projet de classification des races de chiens a été conçue pour garantir des performances optimales du modèle. Cette démarche comprend plusieurs étapes cruciales, chacune contribuant à l'amélioration progressive de la précision et de la capacité du modèle à discriminer efficacement entre les différentes races de chiens, même parmi celles qui présentent des similitudes visuelles.

NETTOYAGE DES DONNEES

La qualité des données est essentielle pour la formation d'un modèle de classification précis. Dans cette optique, une étape préliminaire de nettoyage a été entreprise. Tout d'abord, une sélection stratégique de 15 races de chiens similaires a été réalisée parmi les 120 races initiales. Cette sélection vise à 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.

Ensuite, 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. Cependant, 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.



Et 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.



MISE EN PLACE D'UNE BASELINE

Pour établir un comparatif pertinent avec le modèle YOLOv9, nous avons d'abord entraîné un modèle VGG16 qui servira de référence. Il est crucial de noter la distinction entre ces deux modèles : tandis que YOLOv9 est un algorithme de détection d'objets, le modèle VGG16 est un réseau de neurones profond utilisé principalement pour la classification.

Cependant, notre approche diffère ici de l'utilisation traditionnelle de YOLOv9. Plutôt que de se limiter à la détection d'objets, nous avons exploré la possibilité d'entraîner YOLOv9 à reconnaître les différentes races de chiens, similairement à un modèle de classification. Cette démarche nous permettrait non seulement de reconnaître plusieurs races différentes sur une même image, mais aussi d'appliquer cette reconnaissance dans des flux vidéo.

Cette approche novatrice élargit les capacités de YOLOv9 au-delà de la simple détection d'objets, ouvrant la voie à des applications plus complexes et polyvalentes dans le domaine de la vision par ordinateur. En comparant les performances de YOLOv9 avec celles du modèle VGG16, nous pourrions évaluer son aptitude à remplir cette nouvelle fonctionnalité de reconnaissance des races de chiens.

UTILISATION DU MODELE YOLOV9

À ce stade, notre objectif principal est de comprendre le fonctionnement du modèle YOLOv9, qui est disponible sur GitHub à l'adresse suivante : <https://github.com/WongKinYiu/yolov9>. Avant de pouvoir entraîner le modèle sur notre propre jeu de données, certains prérequis doivent être mis en place, tels qu'un fichier pour décoder les labels au format .yaml ainsi qu'une architecture spécifique dans les données.

Pour faciliter cette mise en place, nous avons suivi le tutoriel détaillé proposé par Aarohi Singla sur GitHub : <https://github.com/AarohiSingla/YOLOv9>. Ce tutoriel fournit des instructions claires et des exemples pratiques pour configurer l'environnement nécessaire et préparer les données afin de pouvoir entraîner efficacement le modèle YOLOv9 sur un jeu de données personnalisé.

A ce moment présent, le DataSet est organisé de la manière suivante :

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

Mais pour pouvoir utiliser YOLOv9 nous devons utiliser l'arborescence suivante :

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

Nous avons développé un code personnalisé pour générer les fichiers textes nécessaires afin de labeliser les images. Ce code a été conçu pour annoter chaque image avec le label correspondant à la race de chien, tout en fournissant les coordonnées des boîtes de détection. L'utilisation de YOLOv9 pour identifier les chiens sur les images (lors du nettoyage) a simplifié cette tâche, car cela nous a permis de recadrer directement sur les chiens, générant ainsi des boîtes de détection qui représentent l'ensemble de l'image.

Cette étape était cruciale pour fournir au modèle les informations nécessaires pour reconnaître et classifier les différentes races de chiens avec précision. Les données ainsi annotées ont ensuite été utilisées pour entraîner le modèle à reconnaître et classifier les différentes races de chiens. Les métriques de performance telles que l'accuracy, la courbe de perte et la matrice de confusion ont été utilisées pour évaluer et ajuster le modèle au fur et à mesure de son apprentissage. Ce processus itératif nous a permis d'améliorer progressivement les performances du modèle et d'optimiser sa capacité à reconnaître les races de chiens avec précision.

OPTIMISATION DU MODELE

Dans la phase d'optimisation du modèle YOLOv9, nous avons suivi une approche progressive en entraînant d'abord le modèle sur des images non détournées, c'est-à-dire sans utilisation de RMBG 1.4, puis sur des images détournées. Cette méthodologie nous a permis d'évaluer l'impact du détournage sur les performances du modèle.

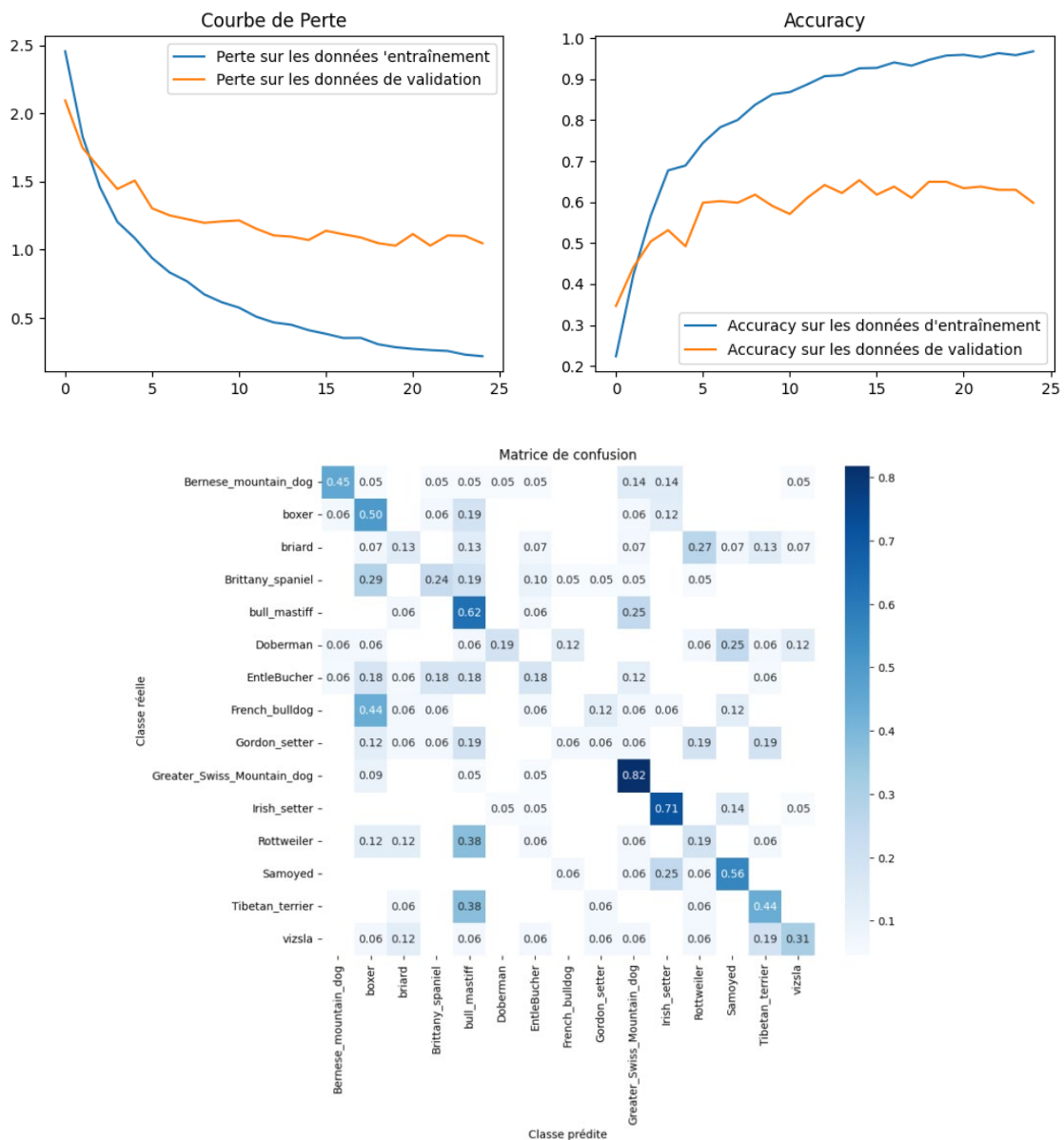
Il était évident que la réduction du bruit sur les images obtenue grâce au détournage a favorisé une meilleure généralisation du modèle YOLOv9 lors de son entraînement. Cette amélioration de la qualité des données d'entrée a conduit à des performances plus stables et plus fiables du modèle.

Par la suite, notre objectif était d'ajuster les hyperparamètres pour obtenir un modèle encore plus performant. Dans le cadre de ce projet de preuve de concept, nous avons observé des résultats prometteurs en augmentant simplement le nombre d'epochs. Cette augmentation a permis au modèle d'apprendre davantage de caractéristiques discriminantes et d'affiner sa capacité à reconnaître et classifier les différentes races de chiens avec précision malgré une baisse de l'accuracy globale.

En combinant l'utilisation de données détournées avec des ajustements d'hyperparamètres, nous avons pu obtenir des améliorations significatives des performances du modèle YOLOv9, démontrant ainsi son potentiel pour la classification précise des races de chiens dans le cadre de notre preuve de concept.

SYNTHESE DES RESULTATS

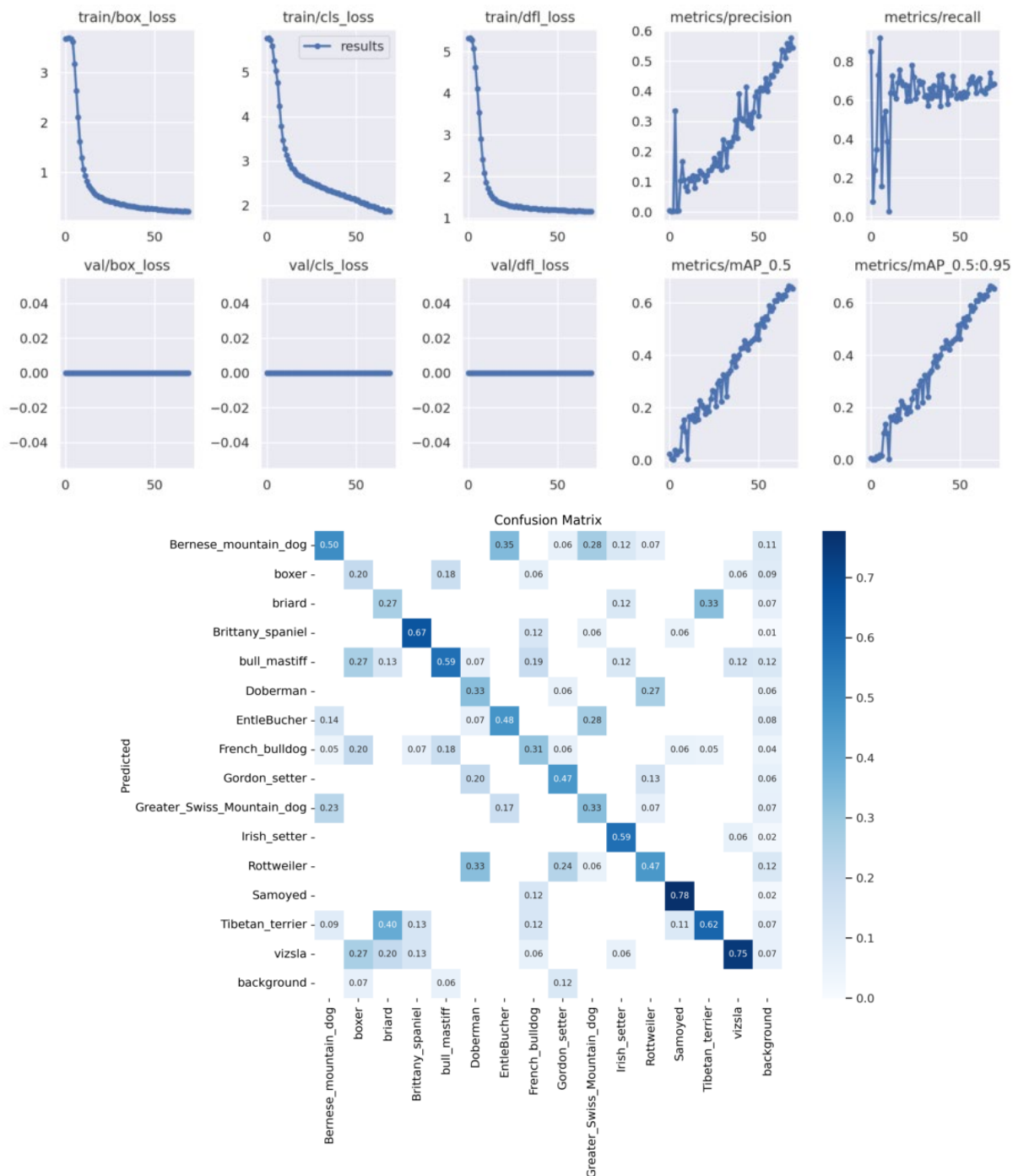
MODELE VGG16



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.

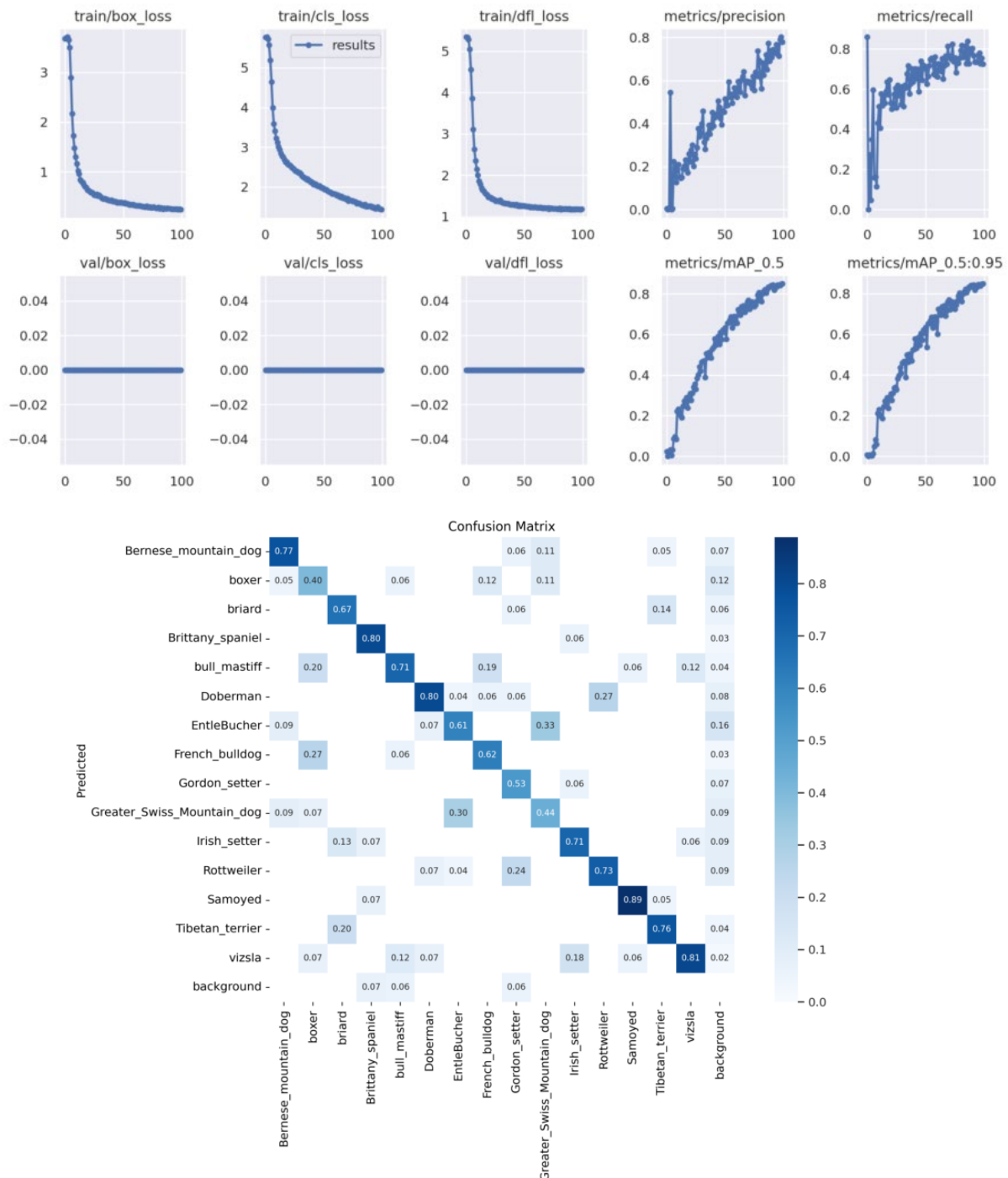
Cette baseline nous fournit un premier score d'accuracy de 0.65 à partir duquel nous allons pouvoir évaluer les performances de YOLOv9, ainsi que l'impact de notre processus de nettoyage préalable.

MODELE YOLOV9 SUR LES IMAGES NON DETOUREES



Le modèle YOLOv9 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.

MODELE YOLOV9 SUR LES IMAGES DETOUREES



En conclusion, notre modèle de classification des races de chiens a atteint une accuracy d'environ 0.73, avec une amélioration notable dans la reconnaissance des races similaires et une amélioration globale des classifications. Ces résultats confirment l'efficacité de notre approche et démontrent le potentiel pratique de notre modèle pour la reconnaissance automatique des races de chiens dans les images et les vidéos.

LES LIMITES ET LES AMELIORATIONS POSSIBLES

Malgré les résultats prometteurs obtenus avec notre approche de modélisation pour la classification des races de chiens, plusieurs limites et pistes d'amélioration demeurent.

Tout d'abord, l'une des principales limites réside dans la taille et la diversité de notre ensemble de données. Bien que notre modèle ait démontré une précision respectable, l'ajout de données supplémentaires provenant de diverses sources pourrait améliorer sa capacité à généraliser à un plus large éventail de races de chiens et de conditions d'imagerie.

En outre, notre modèle actuel peut être limité par sa capacité à traiter des variations importantes dans les images, telles que les changements d'éclairage, les occlusions et les poses variées des chiens. L'intégration de techniques de data augmentation et de régularisation pourrait aider à atténuer ces problèmes en enrichissant notre ensemble de données d'entraînement et en renforçant la robustesse du modèle.

Par ailleurs, une autre limitation potentielle réside dans l'interprétabilité du modèle. Bien que notre modèle soit capable de classifier les races de chiens avec précision, il peut être difficile de comprendre les raisons derrière ses prédictions. L'intégration de techniques d'interprétabilité telles que la visualisation des activations des couches du réseau ou l'utilisation de méthodes d'exploration de modèles pourrait améliorer notre capacité à comprendre comment notre modèle prend ses décisions.

Une autre limite importante à prendre en compte est l'absence de bibliothèques ou de frameworks couramment utilisés qui proposent spécifiquement le modèle YOLOv9. Cette lacune rend non seulement difficile la personnalisation du modèle, mais aussi l'utilisation efficace des hyperparamètres.

En l'absence de bibliothèques dédiées, les utilisateurs doivent souvent recourir à des implémentations personnalisées ou à des adaptations de modèles existants pour intégrer YOLOv9 dans leurs projets. Cela peut entraîner des défis techniques et des contraintes de temps, en particulier pour les utilisateurs moins expérimentés en apprentissage automatique ou en vision par ordinateur.

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 pourrions continuer à affiner notre modèle et à le rendre plus efficace pour la reconnaissance des races de chiens dans diverses applications.