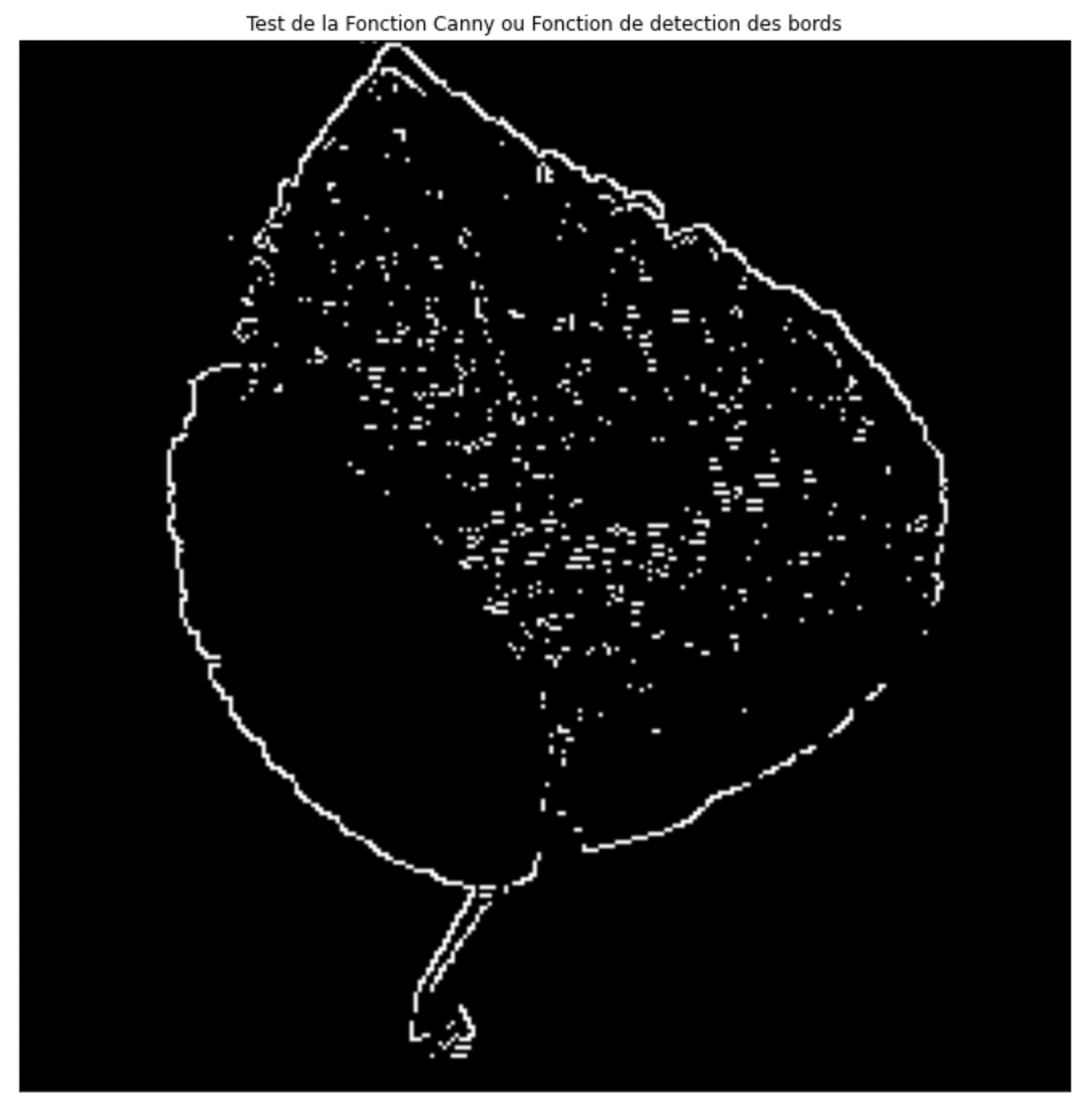
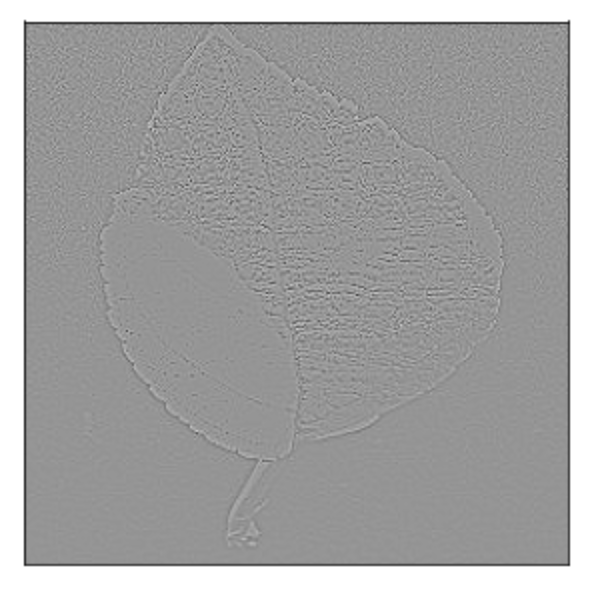
* Exploration de couleur dans les données images de plantes

Enfin, nous terminons avec une étude plus approfondie sur la variation des canaux de couleur dans une image pour une plante saine et l'autre malade.

Celle-ci s’accompagne de la conception de différents calques de segmentation qui nous a permis entre autres d’extraire les contours de l’image, sa texture. Des caractéristiques qui ont été explorées dans le but d'identifier les différentes plantes..



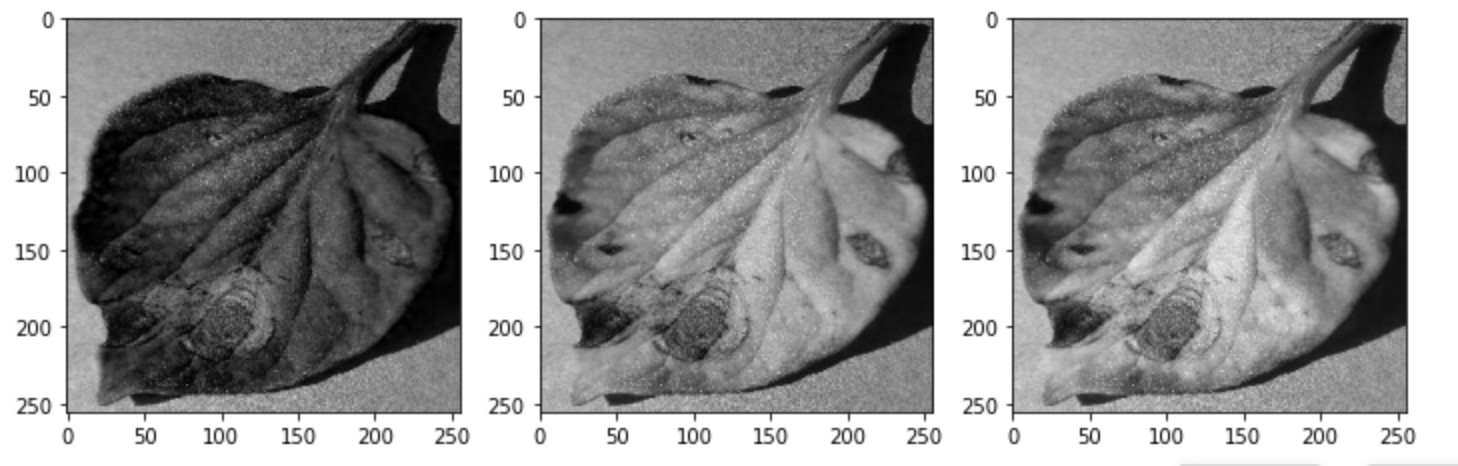
Calque de detection des formes



Calque de detection de la texture

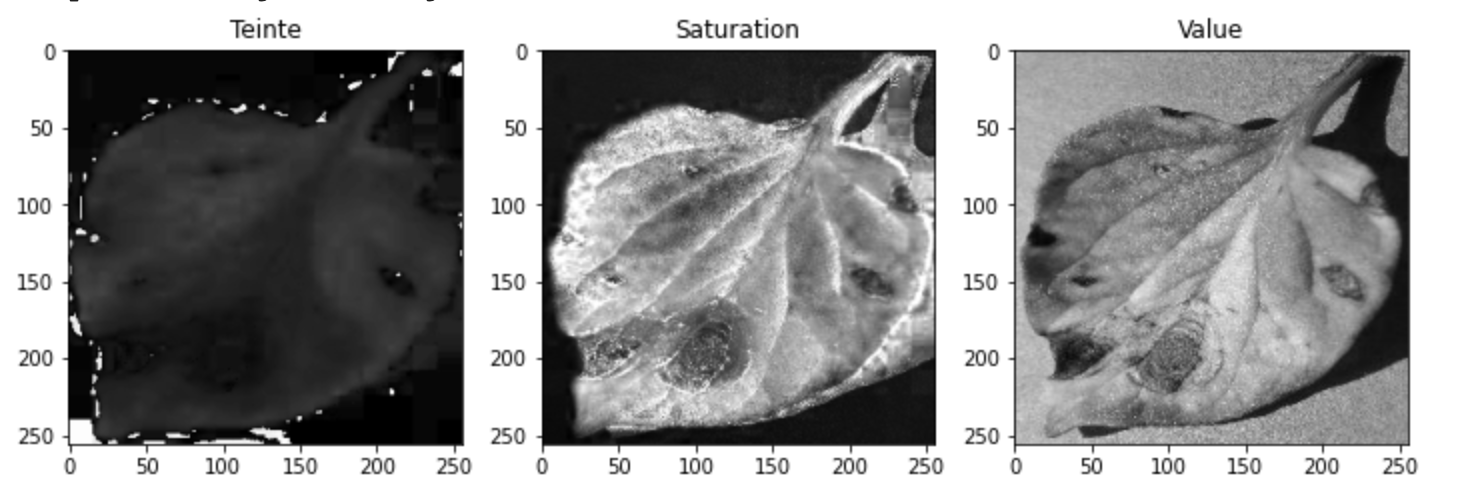
Nous avons remarqué antérieurement que les plantes atteintes de maladies avaient tendance à présenter des couleurs plus jaunes ou marron que les plantes saines. En utilisant la séparation des canaux RVB, nous avons pu mettre en évidence ces différences de teintes et utiliser cette information pour diagnostiquer les maladies.

Nous avons pu noter que les images surexposées ont une moyenne à 255 255 255, est ce une solution pour classer ces images.



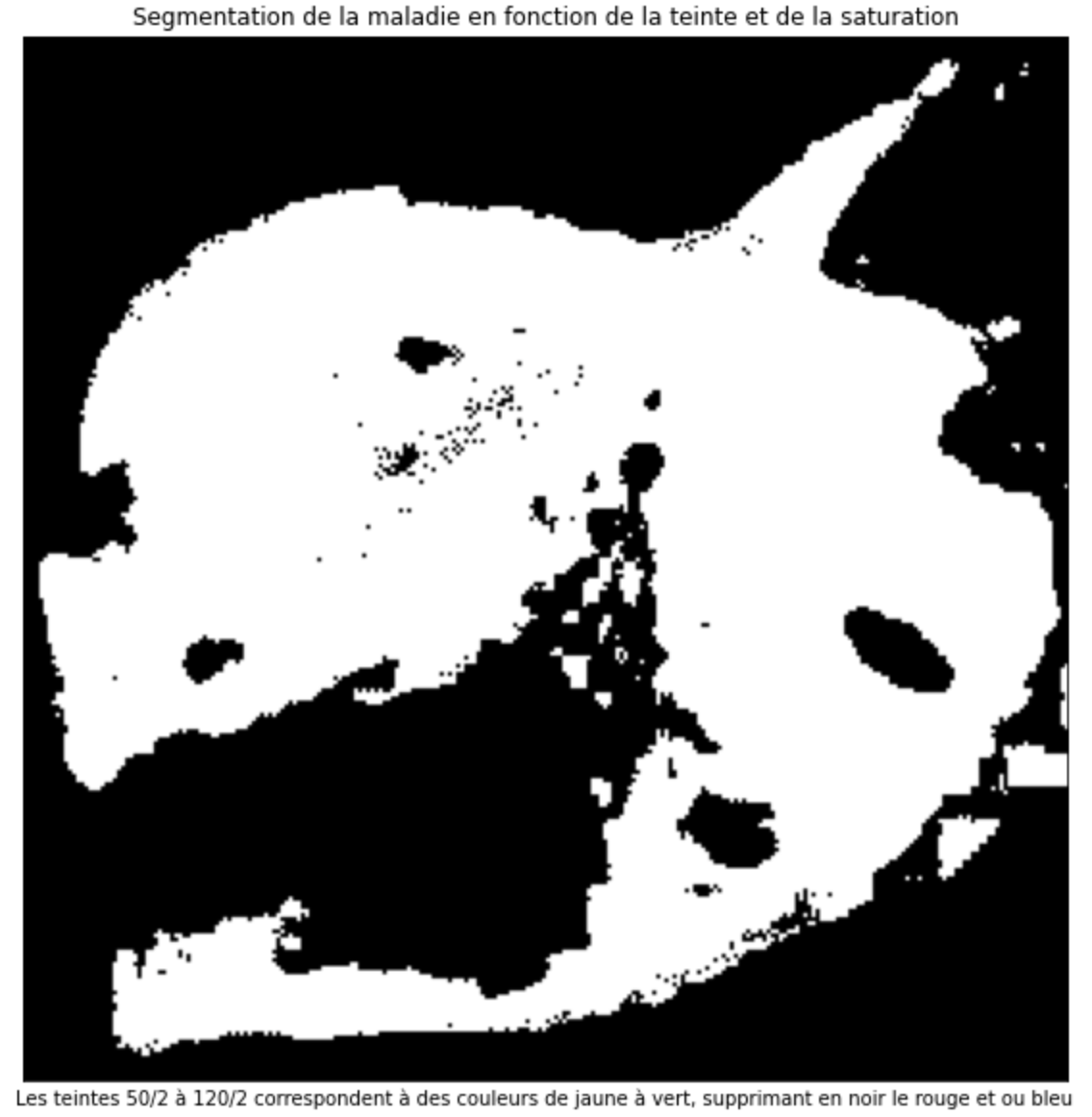
Séparation des couleurs R, V ,B

Nous avons exploré l'utilisation de calques pour extraire les valeurs, les teintes, la saturation ou la balance des blancs des images dans le but d'augmenter notre jeu de données et d'équilibrer les classes pour améliorer les performances de prédiction. Cette démarche avait pour objectif d'améliorer les performances de prédiction en permettant à notre modèle d'analyser plus de cas et de disposer d'une représentation plus précise des différentes classes de données.



Légende

Voici un exemple avec l’utilisation d’un calque pour une image de plante malade, qui a été segmentée en différentes parties pour extraire les zones identifiées comme présentant des symptômes de maladies. Les zones présentant des couleurs anormales ont été isolées, ainsi que les parties affectées par des lésions ou des taches. Ces segments ont été séparés en différentes parties identifiables pour permettre une analyse plus précise de chaque zone et une meilleure évaluation de l'état de la plante.



Légende

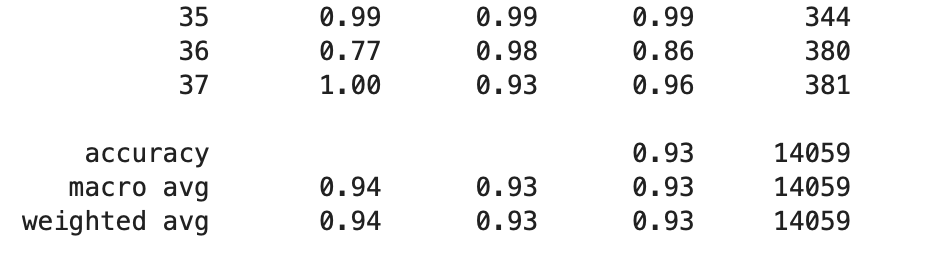
Classification des plantes :

Une première classification à été créé avec les 38 classes, celle-ci sur un modèle VGG16 pour visualiser de premier résultat. Ce modèle préconfiguré repose sur le réseau VGG mais avec une normalisation par lots, ce qui signifie que chaque couche du réseau est normalisée. Le jeu de donnée ImageNet contient plus de dix millions d'URL d'images avec libellés. Un million d'images sont également associées à des cadres de délimitation spécifiant un emplacement plus précis pour les objets étiquetés.



Modèle VGG16

Après avoir utilisé le modèle préconfiguré VGG16, nous avons ajouté des couches Denses pour l'adapter à notre problématique. Après quelques epochs, nous avons observé que le modèle atteignait un plateau avec une précision d'environ 93%.



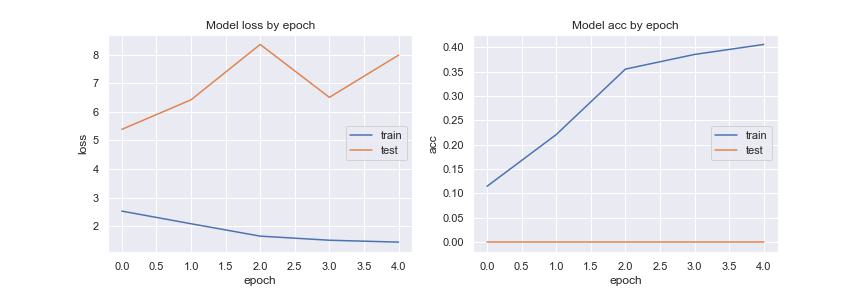
Légende

Nous avons testé différents hyper-paramètres pour optimiser notre modèle, comme l'optimizer 'Adam', qui a finalement été retenu car les autres options n'étaient pas plus convaincantes. Nous avons obtenu une matrice de confusion à la suite de cette prédiction.



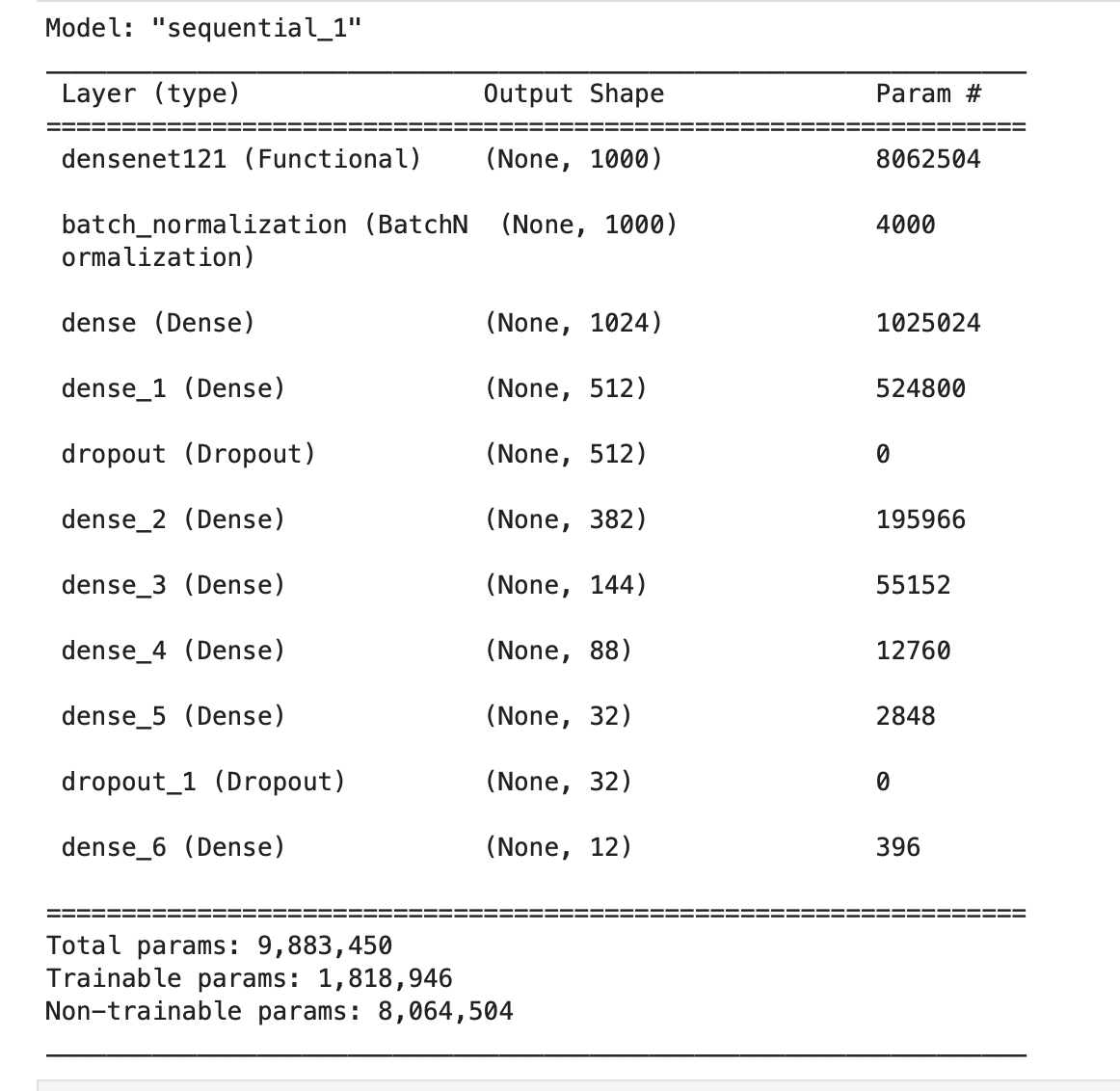
Légende

Nous avons également exploré une autre approche consistant à classer les images selon le type de plante, puis par maladie. Pour cela, nous avons repris notre modèle initial VGG16 et l'avons entraîné sur 14 classes.



Graphique représentatif de l’entrainement du modèle sur quelques épochs

Nous avons conçu différents types de modèles pour chacune des 14 classes, notamment en utilisant le transfert learning avec DenseNet121. Ce modèle préconfiguré est un réseau dense qui a été entraîné sur le jeu de données Imagenet et a une profondeur de 121 couches. Contrairement à RESNET, qui combine les couches à l'aide de l'addition, DenseNet combine les couches à l'aide de la concaténation.

Nous avons principalement utilisé les fonctions de perte 'sparse\_categorical\_crossentropy' et 'categorical\_crossentropy', ainsi que les métriques 'accuracy' et 'acc'. Nous avons également testé différentes fonctions d'activation telles que 'relu' et 'sigmoid', avec une dernière couche en 'softmax'. Cependant, nous avons observé de l'overfitting, et avons donc exploré différents callbacks tels que 'learning-rate' ou 'checkpoint', ainsi que des layers de normalisation, de régularisation et de pooling, notamment ceux inclus dans les modèles préconfigurés.

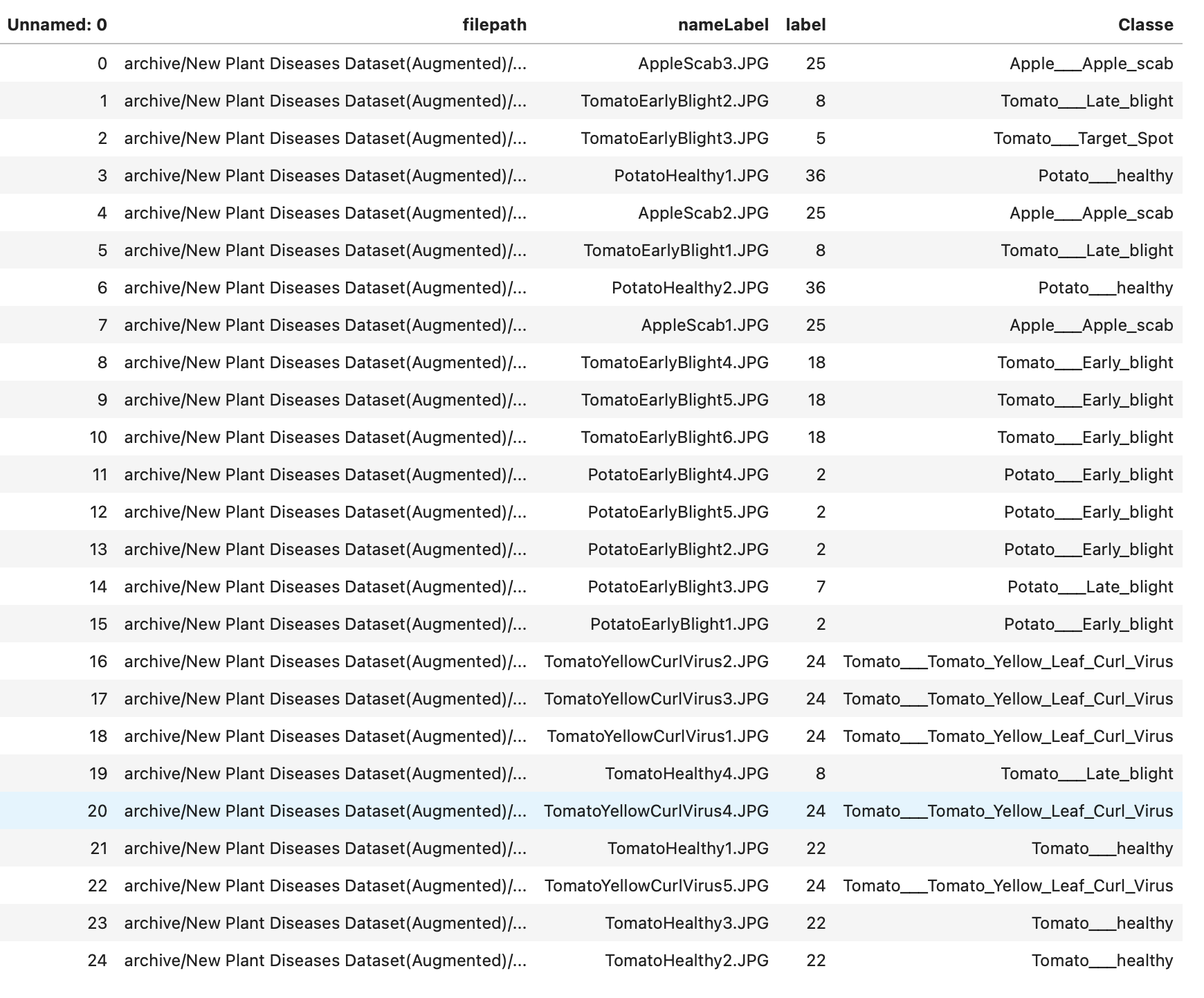
Les couches de convolution ont principalement été en Conv2D ou Dense. Bien que d'autres types de couches telles que SeparableConv2D, Conv3D ou DepthwiseConv2D auraient pu être étudiées, leurs dimensions et paramètres sont peu intuitifs avec des modèles de transfert learning, nous avons donc choisi de les écarter.

| Différents résultats selon par type de classification | | |
| --- | --- | --- |
| Classe de Plante | Modèle utilisé | Graphique de précision et de perte |
| Pomme | InceptionV3 | Image |
| Pomme | Conv2D | Image |
| Tomato | Conv2D | Image |
| Fraise | DenseNet121 | Image |
| Type de plante | InceptionResnetV2 | Image |

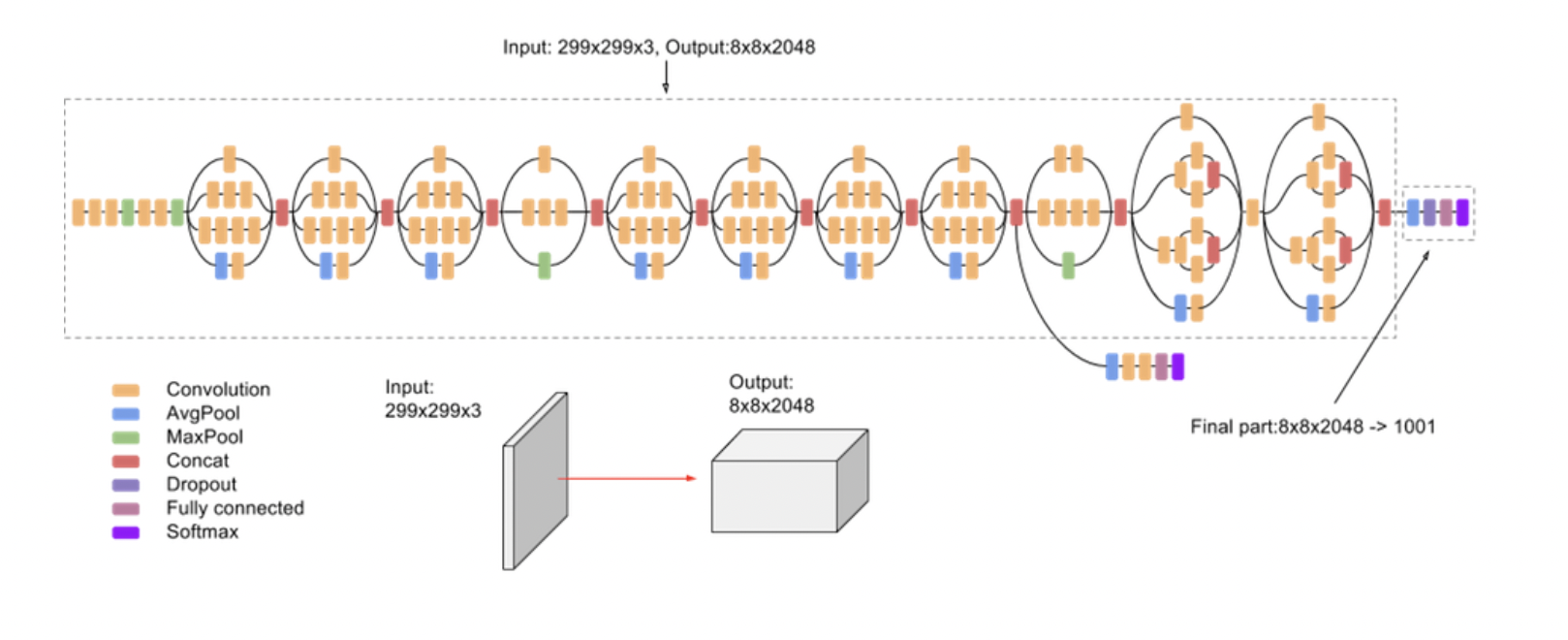
A l’aide de fonctions de visualisation quelques observations de classes peuvent être observé.

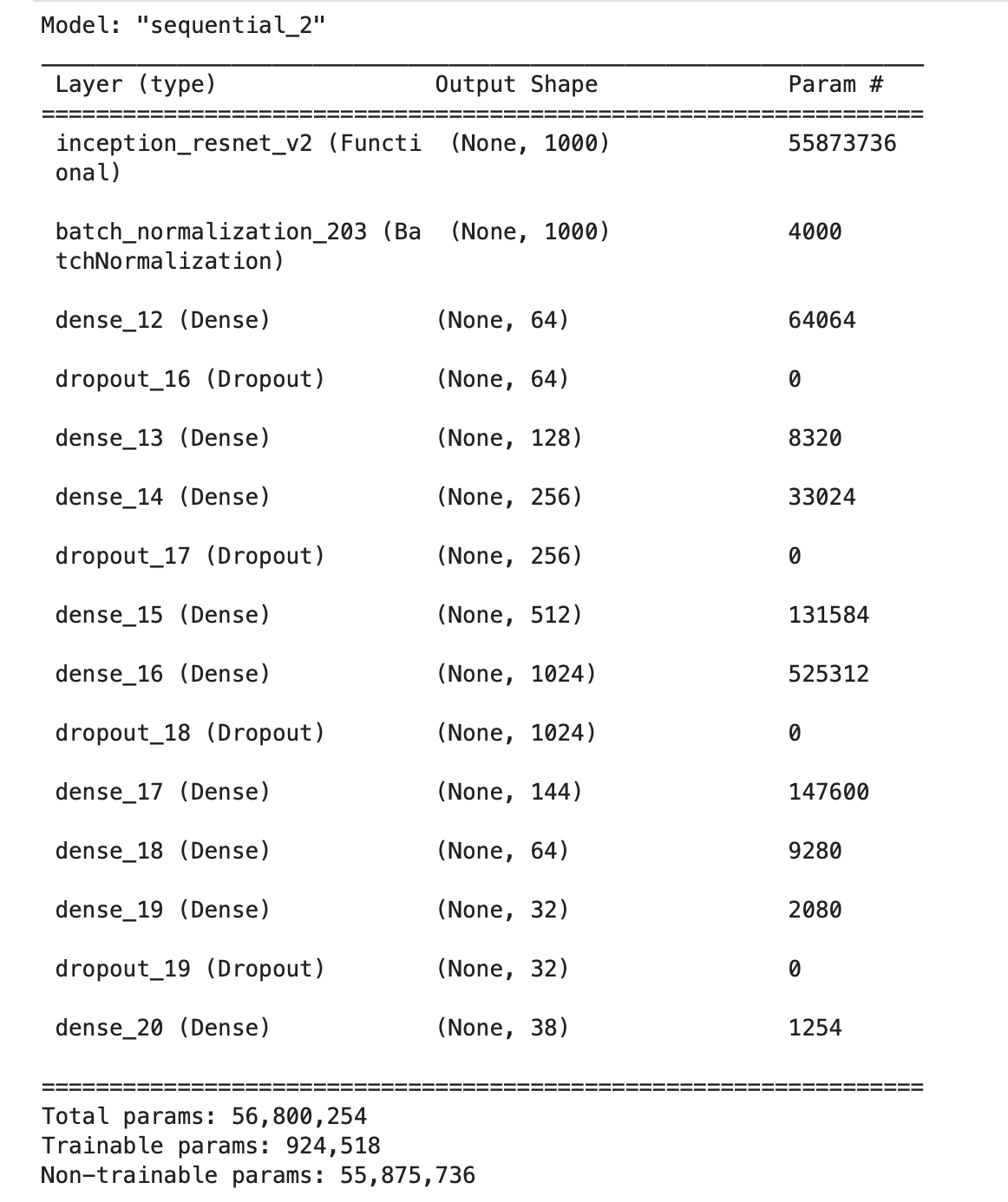
La construction du dataset ci-dessous comprenant les chemins des images, les classes réelles et les prédictions obtenues à partir de nos modèles, ainsi qu'un label unique pour chaque classe, a permis d'évaluer objectivement la performance de nos modèles de classification sur un ensemble de données externe. Cette approche peut valider nos modèles et identifie les erreurs de classification qui ont été corrigées en ajustant les paramètres du modèle ou en révisant les images mal étiquetées.



Des modèles plus complexes ont été commencé à développer avec une architecture plus profonde et complexe que nos modèles initiaux. Ils prennent également plus de temps à entraîner en raison de leur taille et de leur quantité de paramètres. Malgré cela, nous sommes convaincus qu'ils pourraient fournir des prédictions plus précises et plus fiables pour notre classification à long terme. Pour le moment, ces modèles ne sont que des prototypes et nécessitent encore beaucoup de travail et d'optimisation avant d'être déployés en production.

Quelques détails sur les modèles :

Inception v3 est un modèle de reconnaissance d'image qui a atteint une précision supérieure à 78,1% sur l'ensemble de données ImageNet. Ce modèle est l'aboutissement de nombreuses idées développées par plusieurs chercheurs au fil des ans.



Légende

En voici un diagramme :

Le modèle lui-même est composé de composants symétriques et asymétriques, dont les convolutions, le pool moyen, le pool maximal, les concaténations, les abandons et les couches entièrement connectées. La normalisation des lots est utilisée de manière approfondie dans le modèle et appliquée aux entrées d'activation. La perte est calculée à l'aide de Softmax.



Légende