



# **Apprentissage Profond**

## **Rendu Projet - Classification d'Images**

Tom Baran  
Olivier Blot  
Yohann Djafer

Département Sciences du Numérique - Deuxième année  
2023-2024

# Table des matières

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Introduction</b>                                | <b>3</b>  |
| 1.1      | Contexte et Objectifs . . . . .                    | 3         |
| 1.2      | Problématique . . . . .                            | 3         |
| 1.3      | Méthodologie . . . . .                             | 3         |
| <b>2</b> | <b>Réseau Initiale</b>                             | <b>4</b>  |
| 2.1      | Modèle Initial . . . . .                           | 4         |
| 2.1.1    | Méthodologie Employée . . . . .                    | 4         |
| 2.1.2    | Résultats Obtenus . . . . .                        | 4         |
| 2.2      | Transfert Learning . . . . .                       | 4         |
| 2.2.1    | Méthodologie Employée . . . . .                    | 4         |
| 2.2.2    | Résultats Obtenus . . . . .                        | 5         |
| 2.3      | Fine Tuning . . . . .                              | 6         |
| 2.3.1    | Méthodologie Employée . . . . .                    | 6         |
| 2.3.2    | Résultats Obtenus . . . . .                        | 6         |
| <b>3</b> | <b>Approche d'Amélioration</b>                     | <b>8</b>  |
| 3.1      | Amélioration de la Base de Données . . . . .       | 8         |
| 3.1.1    | Résultats Améliorés . . . . .                      | 8         |
| 3.2      | Augmentation de la Résolution des Images . . . . . | 9         |
| 3.2.1    | Résultats Améliorés . . . . .                      | 9         |
| <b>4</b> | <b>Implémentation et Tests</b>                     | <b>11</b> |
| 4.1      | Procédure . . . . .                                | 11        |
| 4.2      | Entraînement du Réseau . . . . .                   | 11        |
| <b>5</b> | <b>Conclusion et Perspectives</b>                  | <b>12</b> |
| 5.1      | Résultats Obtenus . . . . .                        | 12        |
| 5.2      | Limites et Défis . . . . .                         | 12        |
| 5.3      | Perspectives d'Amélioration . . . . .              | 12        |
| <b>A</b> | <b>Annexes</b>                                     | <b>12</b> |

# Table des figures

|   |  |    |
|---|--|----|
| 1 | Resultats du modèle initial . . . . .                          | 4  |
| 2 | Resultats du modèle initial . . . . .                          | 5  |
| 3 | Resultats du modèle initial . . . . .                          | 7  |
| 4 | Resultats après augmentation de la base de données . . . . .   | 9  |
| 5 | Resultats après augmentation de la taille des images . . . . . | 10 |

# 1 Introduction

## 1.1 Contexte et Objectifs

Le projet présenté vise à développer une intelligence artificielle capable de classifier des images de produits de boulangerie, spécifiquement des baguettes, des pains au chocolat et des croissants. La classification précise de ces produits est cruciale, notamment pour des applications telles que l'automatisation dans les boulangeries, l'inventaire automatique ou encore les recommandations personnalisées dans les commerces en ligne. L'objectif principal est d'atteindre une haute précision dans la reconnaissance de ces différents types de viennoiseries afin de minimiser les erreurs de classification et d'améliorer l'efficacité des systèmes automatisés. De plus, une IA japonnaise similaire c'est avéré étonnamment utile dans la reconnaissance de certains types de cellules cancéreuses.

## 1.2 Problématique

Une des principales problématiques rencontrées est la confusion fréquente entre les croissants et les pains au chocolat. Cette confusion est due à des similitudes visuelles entre ces deux viennoiseries, malgré leurs différences distinctives. De plus, une autre contrainte majeure est la limitation des ressources computationnelles, notamment l'accès restreint aux GPU nécessaires pour entraîner des modèles de deep learning efficaces. Cette limitation impacte directement la capacité à expérimenter avec des modèles plus complexes ou à effectuer des entraînements prolongés.

## 1.3 Méthodologie

Pour aborder ces défis, plusieurs approches méthodologiques ont été adoptées :

- **Augmentation de données** : L'augmentation de données a été utilisée pour enrichir le jeu de données d'entraînement. Des techniques telles que la rotation, le zoom, les transformations de couleur et le recadrage ont été appliquées pour générer des variations des images existantes, augmentant ainsi la diversité des données et améliorant la robustesse du modèle.
- **Transfert de learning** : Le transfert de learning implique l'utilisation de modèles pré-entraînés sur des ensembles de données plus vastes et génériques, puis leur adaptation aux spécificités de notre jeu de données. Cette approche permet de tirer parti des connaissances acquises par des modèles complexes sans nécessiter un entraînement complet à partir de zéro.
- **Fine-tuning** : Le fine-tuning consiste à ajuster finement les poids des modèles pré-entraînés sur notre jeu de données spécifique. Cette étape permet d'optimiser les performances du modèle pour la tâche de classification des viennoiseries tout en utilisant efficacement les ressources computationnelles limitées disponibles.

## 2 Réseau Initiale

### 2.1 Modèle Initial

Pour commencer, nous avons réalisé un réseau de neurones en nous inspirant de celui du TP3 (reconnaissance d'images de chien et chat).

#### 2.1.1 Méthodologie Employée

Il commence par une couche de convolution avec 32 filtres de taille 3x3 et une activation ReLU, prenant en entrée des images de taille 128x128 avec trois canaux de couleur (RVB). Cette couche est suivie d'une couche de max pooling pour réduire les dimensions spatiales. Le modèle continue avec une deuxième couche de convolution de 64 filtres 3x3, également suivie d'une couche de max pooling. Une troisième couche de convolution avec 96 filtres et une quatrième avec 128 filtres, chacune suivie de max pooling, permettent d'extraire des caractéristiques de plus en plus complexes. Les sorties sont ensuite aplaties par une couche Flatten, transformant le tenseur en un vecteur. Ce vecteur est passé à une couche dense comportant 512 neurones avec activation ReLU pour capturer les relations non linéaires. Enfin, une couche de sortie avec une activation softmax produit des probabilités pour les trois classes : baguette, pain au chocolat et croissant.

Le modèle est compilé avec la fonction de perte sparse categorical crossentropy et l'optimiseur Adam avec un taux d'apprentissage de 0.0003, et il est évalué en termes de précision catégorielle sparse. Le modèle a été entraîné sur 50 époques avec un batch size de 10, utilisant des ensembles de données de validation pour évaluer les performances pendant l'entraînement.

#### 2.1.2 Résultats Obtenus

- **Précision sur l'Ensemble de Validation** : La précision est de 50%.
- **Perte sur l'Ensemble de Validation** : La perte atteint 1 (avec 10 epochs).

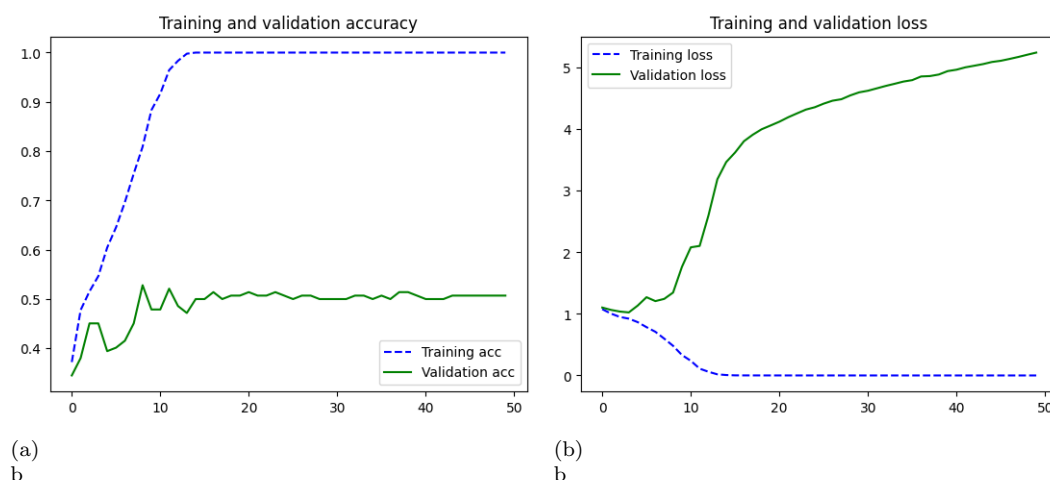


FIGURE 1 – Résultats du modèle initial

La précision sur l'ensemble de validation avec ce premier modèle était assez faible 50% (au même stade c'était 70% pour le tp3). De plus on remarque qu'il y a rapidement du surapprentissage (10 epochs) ce qui pourrait s'expliquer par le fait que nous avons une petite base d'apprentissage d'environ 1200 images. Dans le TP3, le surapprentissage est moins conséquent cependant la base d'apprentissage est plus grande (2000 images).

## 2.2 Transfert Learning

### 2.2.1 Méthodologie Employée

- **Modèle Pré-entraîné** : Nous avons utilisé le modèle pré-entraîné VGG16 sur l'ensemble de données ImageNet, un vaste ensemble de données contenant des millions d'images classées en

milliers de catégories. Ces couches sont responsables de l'extraction des caractéristiques basiques et complexes des images.

- **Gel des Couches Initiales** : Les couches convolutives du modèle pré-entraîné ont été "congelées" (leurs poids n'ont pas été mis à jour pendant l'entraînement) pour conserver les caractéristiques apprises.
- **Ajout de Couches Spécifiques** : nous avons ajouté plusieurs couches entièrement connectées au sommet du réseau pré-entraîné VGG16 pour adapter le modèle à notre tâche de classification.

La première couche ajoutée est une couche de flattening, qui transforme les sorties 3D de la dernière couche convolutive de VGG16 en un vecteur 1D.

Ensuite, une couche dense avec 256 neurones et une activation relu a été ajoutée. Cette couche permet de capter des caractéristiques complexes et de haut niveau nécessaires pour la classification.

Enfin, une couche dense (Dense) finale avec 3 neurones a été ajoutée. Cette couche utilise une activation softmax pour produire les probabilités d'appartenance aux trois classes cibles de notre tâche de classification.

## 2.2.2 Résultats Obtenus

- **Précision sur l'Ensemble de Validation** : La précision est de 65%.
- **Perte sur l'Ensemble de Validation** : La perte est de 0,8.

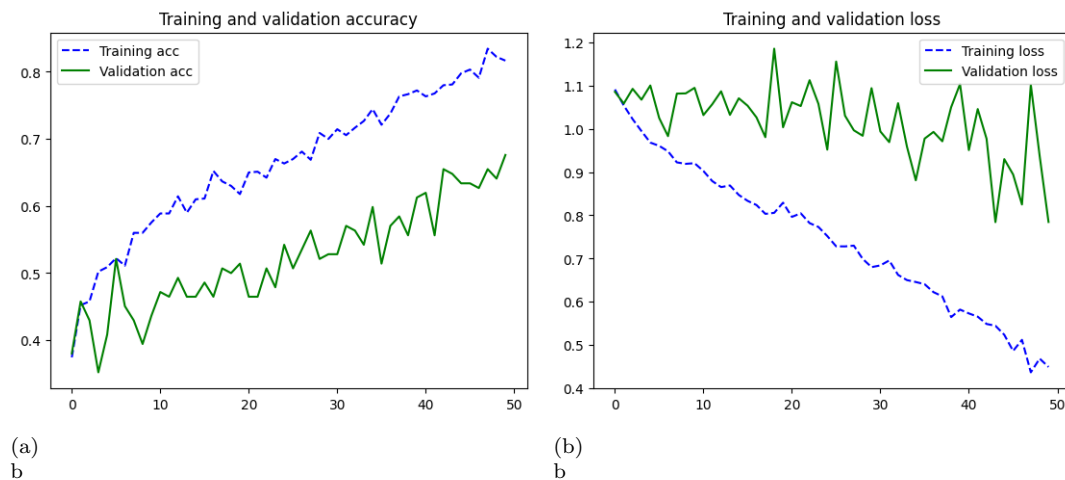


FIGURE 2 – Resultats du modèle initial

Le transfert learning nous a permis d'utiliser un modèle pré-entraîné sur un large ensemble de données, nous avons donc pu obtenir une précision plus élevée qu'avec notre modèle et notre base de données. Cette méthode permet d'optimiser les ressources et de réduire le temps d'entraînement tout en atteignant une performance élevée, démontrant ainsi l'efficacité du transfert learning pour des tâches de classification d'images avec des ensembles de données limités.

## 2.3 Fine Tuning

### 2.3.1 Méthodologie Employée

1. **Initialisation avec un Modèle Pré-entraîné** : Un modèle pré-entraîné sur une vaste base de données est utilisé comme point de départ. Ici nous utilisons VGG16.
2. **non-Gel des Couches Initiales** : Les premières couches du réseau ne sont pas gelées ce qui permet de d'affiner la précision du modèle pré-entraîné.
3. **Ajout de Couches Spécifiques** : nous avons ajouté plusieurs couches entièrement connectées au sommet du réseau pré-entraîné VGG16 pour adapter le modèle à notre tâche de classification.

La première couche ajoutée est une couche de flattening, qui transforme les sorties 3D de la dernière couche convolutive de VGG16 en un vecteur 1D.

Ensuite, une couche dense avec 256 neurones et une activation relu a été ajoutée. Cette couche permet de capter des caractéristiques complexes et de haut niveau nécessaires pour la classification.

Enfin, une couche dense finale avec 3 neurones a été ajoutée. Cette couche utilise une activation softmax pour produire les probabilités d'appartenance aux trois classes cibles de notre tâche de classification.

4. **Réentraînement des Couches Finales** : le modèle (vgg16 + couches spécifiques) est réentraîné avec un taux d'apprentissage plus faible.

### 2.3.2 Résultats Obtenus

Le fine tuning a montré une amélioration significative par rapport au modèle initial non ajusté. Les résultats sont les suivants :

- **Précision sur l'Ensemble de Validation** : La précision est de 85%.
- **Perte sur l'Ensemble de Validation** : La perte est de 0.8.
- **Précision sur l'ensemble de test** : 91%.
- **Perte sur l'ensemble de test** : 0.35

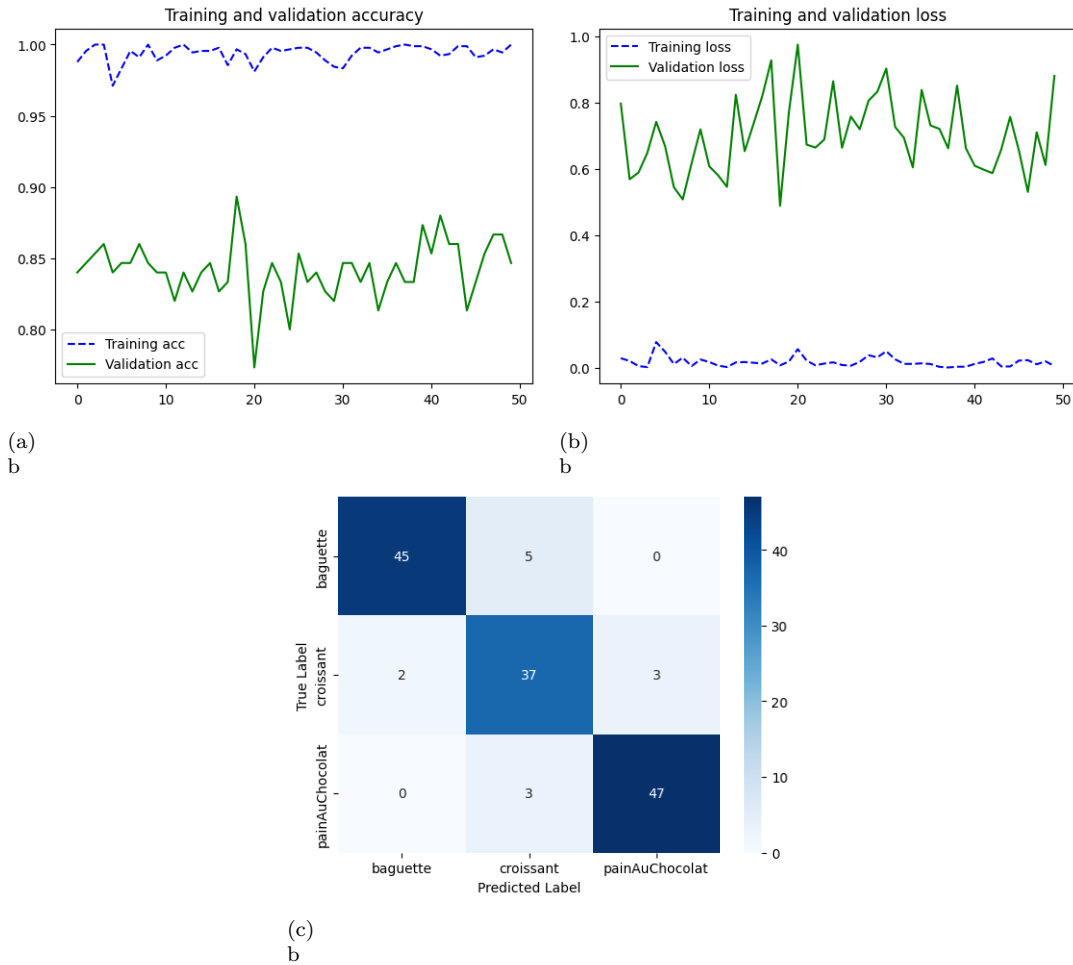


FIGURE 3 – Resultats du modèle initial

Nous remarquons alors que le modèle avec fine tuning est plus précis que le modèle avec transfer learning sur les données de validation.

Ces résultats sont à nuancés notamment car nos images de validation et d'entraînement sont proches et le modèle pré-entraînée avec le transfer learning n'est pas entraînée avec nos images.

Le modèle avec fine tuning donne de bon résultats sur les données de test avec 90% de précision.

La matrice de confusion nous montre que le modèle reconnaît très bien les baguettes mais a plus de mal à distinguer les croissants des baguettes.

En relançant les test plusieurs fois le réseau confond souvent plus souvent un croissant avec un pain au chocolat q'un pain au chocolat avec un croissant

En conclusion, le fine tuning s'est révélé être une stratégie efficace pour améliorer la performance du modèle de classification d'image, en utilisant les avantages des connaissances préalablement acquises par un modèle entraîné sur une vaste base de données, tout en adaptant les capacités du modèle aux spécificités de notre tâche cible.

## 3 Approche d'Amélioration

### 3.1 Amélioration de la Base de Données

Les résultats précédents sont obtenus avec une base d'apprentissage, de test et de validation dont les données ont été arbitrairement choisies. Nous pouvons essayer de voir si nous obtenons de meilleurs résultats en modifiant nos jeux de données selon certains critères, en particulier en ce qui concerne les croissants et les pains au chocolat. En effet, comme nous avons pu le constater, le réseau confond souvent un croissant avec un pain au chocolat, mais confond moins un pain au chocolat avec un croissant.

Nous avons donc modifié les données d'apprentissage des pains au chocolat pour faire ressortir davantage leurs éléments distinctifs. Notamment, certaines images de pains au chocolat contenaient des croissants. Ces images ont été retirées de la nouvelle base d'apprentissage.

Un autre point important est que la base de données initiale comportait 300 données d'apprentissage par label, 50 de test et 50 de validation, sauf pour le croissant qui n'avait que 42 données de validation. Les données de test et de validation ont été ajustées pour avoir 50 données de validation pour les croissants.

Est-ce que ces changements vont améliorer le réseau ? L'idée derrière cette amélioration est de fournir des données plus propres et mieux équilibrées, ce qui devrait théoriquement permettre au modèle de mieux distinguer les croissants des pains au chocolat et d'augmenter la précision globale de la classification.

#### 3.1.1 Résultats Améliorés

- **Précision sur l'Ensemble de Validation : 94%**
- **Précision sur l'ensemble de test : 92%**

En conclusion, la modification de la base de données d'apprentissage et de validation a permis d'augmenter nettement la précision de notre modèle. De nombreuses images dans la base de données étaient en double ou trop éloignées d'une viennoiserie classique, ce qui affectait le modèle.



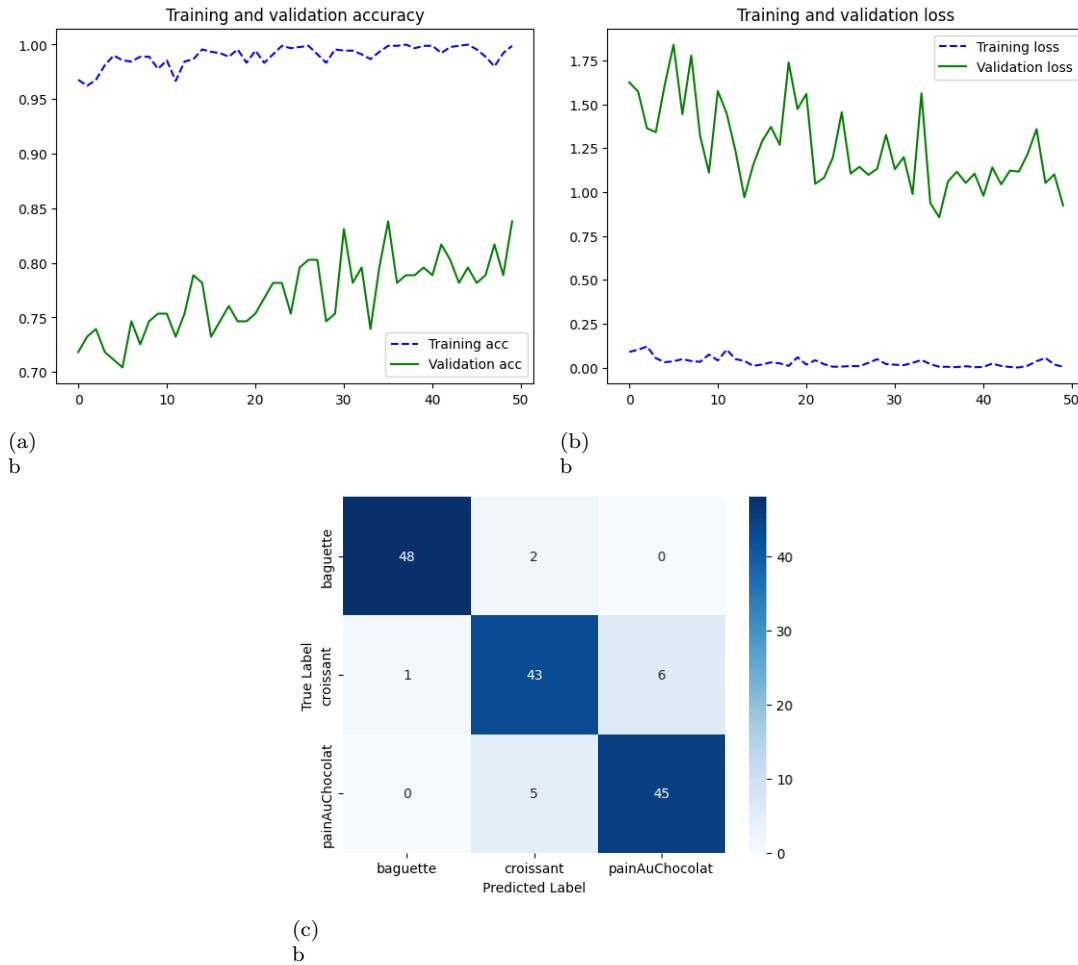


FIGURE 4 – Resultats après augmentation de la base de données

### 3.2 Augmentation de la Résolution des Images

À l'issue de l'apprentissage avec cette nouvelle base de donnée, la précision globale de notre modèle était d'environ 84%, un score satisfaisant compte tenu de la résolution des images de 128x128 pixels. Cependant, une précision plus élevée aurait été souhaitable. Notamment, nous avons observé que notre modèle détectait très bien les baguettes et les pains au chocolat, avec des taux de prédiction de 96% et 90% respectivement. Le principal point de faiblesse était le taux de prédiction des croissants, qui n'était que de 86%. Cela signifie qu'une image de croissant sur trois n'était pas correctement reconnue, souvent confondue avec des pains au chocolat en raison de leur forme similaire.

Pour améliorer ce résultat, nous avons décidé d'augmenter la résolution des images de 128x128 pixels à 256x256 pixels. Cette augmentation de la résolution permet au réseau de neurones d'extraire des caractéristiques plus fines et plus détaillées des images, ce qui est particulièrement utile pour différencier des objets visuellement similaires comme les croissants et les pains au chocolat.

Les résultats après l'augmentation de la résolution des images ont montré une amélioration significative. Les métriques d'apprentissage et de validation se sont améliorées, indiquant une meilleure performance globale du modèle. En particulier, les courbes de précision et de perte pour les phases d'entraînement et de validation montrent une convergence plus rapide et des valeurs finales plus optimales comparées à celles obtenues avec les images en 128x128 pixels.

#### 3.2.1 Résultats Améliorés

- **Précision sur l'Ensemble de Validation** : La précision est de 85%.
- **Perte sur l'Ensemble de Validation** : La perte est de 1.
- **Précision sur l'ensemble de test** : 94%.

- **Perte sur l'ensemble de test** : 0.35
- **Accuracy d'entraînement et de validation** : Les courbes montrent une précision plus élevée et plus stable tout au long de l'entraînement avec les images en 256x256 pixels.
- **Loss d'entraînement et de validation** : La perte a diminué plus rapidement et a atteint des valeurs plus basses, indiquant une meilleure adaptation du modèle aux données.
- **Taux de rappel des croissants** : Le taux de rappel des croissants a augmenté ainsi que les croissants et pain aux chocolat, réduisant les confusions avec les pains au chocolat et améliorant ainsi la précision globale.

Finalement, l'augmentation de la résolution des images a permis une meilleure discrimination des classes, notamment pour les croissants, et a conduit à une amélioration notable des prédictions finales. Cette approche s'est avérée être une piste d'amélioration efficace pour notre projet de reconnaissance d'image.

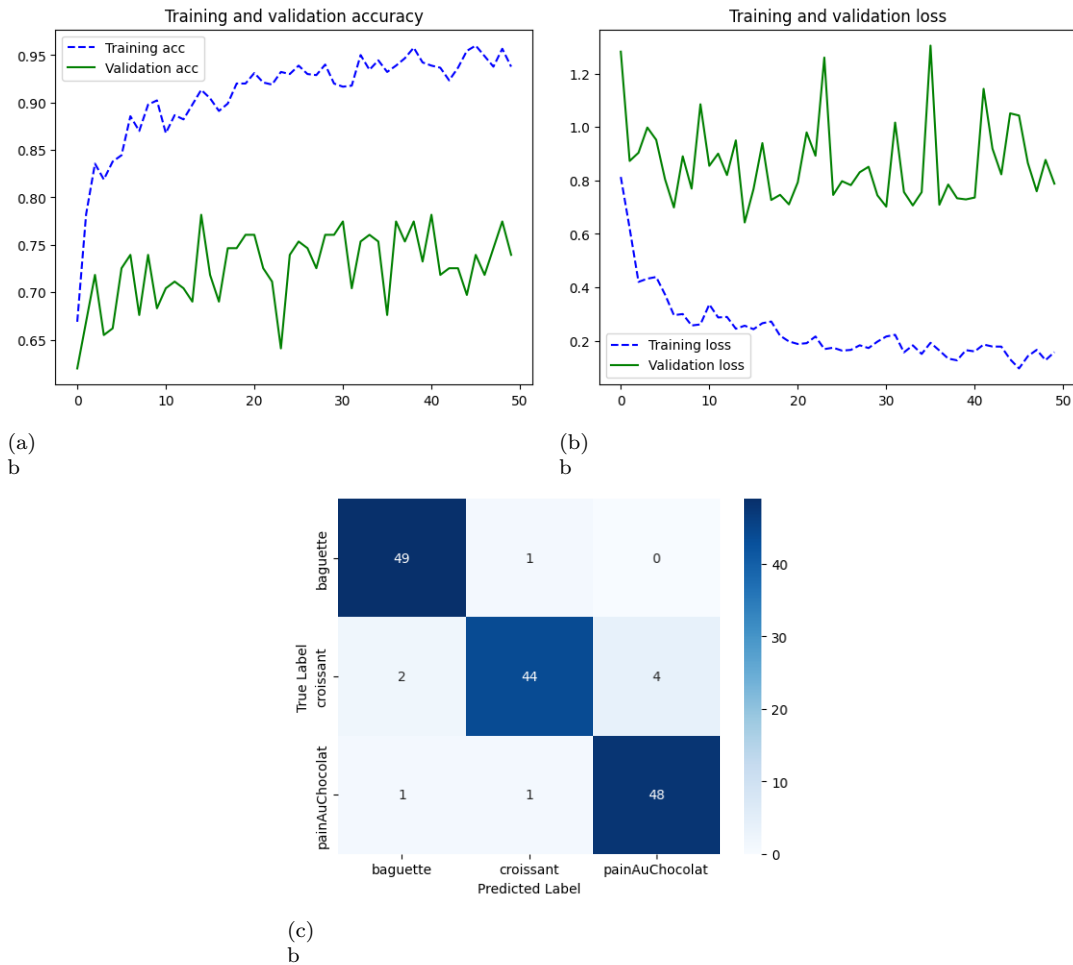


FIGURE 5 – Resultats après augmentation de la taille des images

## 4 Implémentation et Tests

### 4.1 Procédure

Pour reproduire les résultats et tester les modifications, nous avons suivi les étapes suivantes :

1. **Clonage du repository GitHub** : Nous avons commencé par cloner le repository contenant le code et les données du projet.
2. **Importation des modules nécessaires** : Nous avons importé les bibliothèques et les modules nécessaires à l'exécution du projet.
3. **Chargement et analyse des données** : Nous avons exécuté les fonctions de chargement et d'analyse des données pour préparer les jeux de données et effectuer une première exploration des données disponibles.
4. **Exécution des cellules de code** : Nous avons ensuite exécuté les cellules de code à partir de la section III du notebook, où les modèles sont définis et entraînés.

### 4.2 Entraînement du Réseau

L'entraînement du réseau de neurones a été réalisé en utilisant les données modifiées. Les performances du modèle ont été vérifiées à l'aide de diverses métriques, et les résultats ont été comparés à ceux obtenus initialement.

1. **Utilisation des données modifiées** : Les images de la nouvelle base de données ont été utilisées pour entraîner le modèle. Cela incluait des données d'apprentissage mieux équilibrées et nettoyées pour éviter les confusions entre les classes, notamment entre les croissants et les pains au chocolat.
2. **Vérification des performances** : Les performances du modèle ont été évaluées à l'aide de métriques telles que la matrice de confusion, la précision (accuracy), le taux de rappel (recall), la précision (precision) et le score F1. Ces métriques nous ont permis de mesurer l'efficacité du modèle à différencier les classes.
3. **Comparaison des résultats** : Les résultats obtenus avec les données modifiées ont été comparés aux résultats initiaux pour évaluer les améliorations en termes de précision et de taux de rappel. En particulier, nous avons observé une amélioration notable du taux de rappel des croissants, ce qui a contribué à une meilleure performance globale du modèle.

Ces étapes nous ont permis de vérifier l'impact des modifications apportées aux données sur les performances du modèle et de confirmer l'efficacité des améliorations proposées.

## 5 Conclusion et Perspectives

### 5.1 Résultats Obtenus

Les modifications apportées à la base de données ainsi que l’augmentation de la résolution des images ont permis d’améliorer significativement les performances de notre modèle de reconnaissance d’images. Après ces ajustements, notre IA atteint désormais des taux de précision d’environ 95%. L’augmentation de la résolution des images de 128x128 à 256x256 a particulièrement contribué à une meilleure identification des caractéristiques distinctives de chaque classe, réduisant ainsi les erreurs de classification.

Les améliorations apportées à la base de données, incluant l’élimination des images ambiguës et l’équilibrage des données de validation, ont également joué un rôle crucial dans l’amélioration des résultats. Ces méthodes ont permis de réduire les confusions entre les différentes classes d’images, ce qui se reflète dans les métriques de performance globales et dans la matrice de confusion. Ainsi, l’ensemble des méthodes employées s’est avéré efficace pour augmenter la précision et la robustesse de notre modèle.

### 5.2 Limites et Défis

Malgré les améliorations, plusieurs défis et limites subsistent. L’un des principaux problèmes rencontrés est lié aux ressources computationnelles disponibles, en particulier l’accès limité aux GPU. Cela a restreint la capacité à effectuer des entraînements plus longs ou à utiliser des modèles plus complexes.

De plus, la qualité et la quantité des données disponibles ont un impact significatif sur les performances du modèle. Bien que des efforts aient été faits pour nettoyer et équilibrer les données, des variations dans la qualité des images et la représentativité des classes peuvent encore affecter les résultats.

### 5.3 Perspectives d’Amélioration

Pour aller au-delà des résultats actuels, plusieurs pistes d’amélioration peuvent être explorées. L’utilisation d’autres techniques d’augmentation de données, telles que la rotation, le zoom, ou les transformations de couleur, pourrait augmenter la diversité des données d’entraînement et améliorer la robustesse du modèle.

De plus, l’exploration de modèles plus complexes pourraient offrir des gains supplémentaires en précision. Enfin, la collaboration pour obtenir plus de ressources GPU ou l’utilisation de services cloud plus performants permettrait d’exécuter des entraînements plus longs et d’explorer des architectures de modèles plus avancées, ce qui pourrait encore améliorer les performances de notre réseau de neurones.

## A Annexes

GitHub : [https://github.com/TomBaran501/BDD\\_Boulangerie.git](https://github.com/TomBaran501/BDD_Boulangerie.git)