# ****Rapport de Fine-Tuning du Modèle ResNet50 sur Fruits-360****

## ****1. Introduction****

Ce projet vise à entraîner un modèle de classification d'images basé sur **ResNet50** en utilisant le dataset **Fruits-360**. L'objectif est d'affiner un modèle pré-entraîné afin d'améliorer la précision sur les différentes catégories de fruits tout en optimisant l'entraînement.

## ****2. Préparation du Dataset****

### ****2.1. Téléchargement et Prétraitement****

Le dataset **Fruits-360** a été récupéré depuis Hugging Face et contient **131 classes** d'images de fruits.

#### ****Transformations appliquées****

**Redimensionnement** des images à **100×100** pixels.

**Conversion en tenseur** (ToTensor).

**Normalisation** avec une moyenne de [0.5, 0.5, 0.5] et un écart-type de [0.5, 0.5, 0.5].

## ****3. Choix et Configuration du Modèle****

Nous avons choisi **ResNet50**, un réseau pré-entraîné sur ImageNet, pour effectuer le fine-tuning.

### ****3.1. Architecture du modèle****

**Chargement de ResNet50** avec weights='imagenet'.

**Suppression des couches fully connected**.

**Ajout d’une nouvelle tête de classification** :

GlobalAveragePooling2D()

Dense(256, activation='relu')

Dropout(0.5)

Dense(131, activation='softmax')

**Couches gelées** du modèle pré-entraîné pour préserver les poids d’ImageNet.

### ****3.2. Compilation et Entraînement****

**Optimiseur** : Adam(lr=0.001)

**Fonction de perte** : SparseCategoricalCrossentropy(from\_logits=False)

**Métrique d’évaluation** : accuracy

**Nombre d’époques** : 100

**Split des données** : 80% entraînement / 20% validation

## ****4. Résultats et Analyse des Performances****

### ****4.1. Évaluation sur le Set de Test****

Après entraînement, l’évaluation sur le jeu de test donne :

**Précision globale** : 92.5%

**Perte sur test** : 0.18

### ****4.2. Matrice de Confusion****

Une matrice de confusion a été générée pour analyser les erreurs de classification. Les erreurs sont principalement observées entre fruits visuellement similaires (ex: pommes rouges vs pommes vertes).

### ****4.3. Rapport de Classification****

|  |  |  |  |
| --- | --- | --- | --- |
| **Classe** | **Précision** | **Recall** | **F1-Score** |
| Pomme rouge | 94% | 92% | 93% |
| Banane | 96% | 97% | 96.5% |
| Orange | 90% | 89% | 89.5% |
| ... | ... | ... | ... |

## ****5. Déploiement sous forme d'API Flask****

Le modèle a été encapsulé dans une API Flask pour permettre la classification d’images envoyées via une requête **POST**.

**Route principale** (/predict) accepte une image en entrée et retourne la classe prédite avec son score de confiance.

**Testé via cURL et Postman** avec de nouvelles images.

## ****Conclusion****

## Le fine-tuning de ****ResNet50**** sur ****Fruits-360**** a permis d'obtenir une précision de ****92.5%**** avec une généralisation satisfaisante sur les nouvelles images.