

UNIVERSITÉ PARIS CITÉ



Modèle de Deep Learning pour la détection des déchets dans une séquence vidéo avec Github et Docker

REMMACH Fatima Zahra
MENKAOUNOU Ikram
M'CHICHI Boutayna

Ecandré par :Pr.

M2 MLSD

Janvier 2023

TABLE DES MATIÈRES

1	Introduction	4
2	Contexte du projet et problématique	5
3	Méthodologie	6
1	Description de la base de données utilisée	6
2	Description de la méthode d'apprentissage en profondeur utilisée	7
2.1	Xception	7
2.2	YOLOv3	7
3	Description du déploiement du modèle avec Docker et GitHub	7
4	Résultats	9
1	Présentation des résultats de l'entraînement du modèle	9
2	Evaluation du modèle sur des données de test	9
5	Discussion et conclusion	10
6	Annexes	11

CHAPITRE 1

INTRODUCTION

Docker et GitHub sont des outils populaires pour le développement et le déploiement de logiciels. Docker permet de créer et exécuter des applications dans des conteneurs, ce qui rend le déploiement de ces applications plus facile et plus portable. Quant à GitHub, il s'agit d'un service de contrôle de version et de collaboration qui permet aux équipes de travailler ensemble sur des projets de manière efficace.

Dans ce projet, nous allons montrer comment déployer un modèle de gestion des déchets à l'aide de l'apprentissage en profondeur, et ce, en utilisant Docker et GitHub.

Nous allons donc commencer par créer un modèle en utilisant une base de données de déchets, puis nous allons le déployer dans un conteneur Docker. Ensuite, nous allons utiliser GitHub pour partager le code et le modèle avec d'autres membres de l'équipe, et nous allons configurer un outil d'intégration pour automatiser le processus de déploiement.

CHAPITRE 2

CONTEXTE DU PROJET ET PROBLÉMATIQUE

Dans le cadre de l'intelligence artificielle et de la vision par ordinateur, le suivi d'objet est une tâche visuelle visant à identifier et localiser automatiquement un ou plusieurs objets d'intérêt dans des vidéos ou des séquences d'images, et à former une ou plusieurs trajectoires dans le temps. Le résultat du suivi est important pour la reconnaissance et l'analyse du comportement des objets. Le suivi des objets est une étape clé de l'analyse vidéo et est pertinent dans les tâches visuelles telles que la vidéosurveillance automatisée, l'interaction homme-ordinateur, le système de véhicule sans pilote / autonome, la compression vidéo, etc

Les déchets solides municipaux (DSM) sont l'un des types de déchets post consommation qui comprend les déchets, les biens et produits en fin de vie, les emballages usagés, etc. L'augmentation continue de la production de DSM est associée à une croissance démographique ainsi qu'à une augmentation de la quantité de déchets générée par personne, en particulier parmi les citoyens, ce qui augmente l'importance de la construction d'une chaîne de processus efficace pour l'élimination des DSM. Dans cette histoire, Nous décrirons comment nous avons créé un modèle de vision par ordinateur capable de détecter des déchets sur une vidéo ainsi les classer pour les recycler.

CHAPITRE 3

MÉTHODOLOGIE

Dans ce qui suit, nous allons décrire les étapes suivies pour réaliser le projet, y compris les outils utilisés et les sources de données.

1 Description de la base de données utilisée

Nous avons utilisé la base de données fournie par Andrea Santoro sur Kaggle, accessible sur l'adresse [Lien de la base de données](https://www.kaggle.com/asdasdasdasdas/garbage-classification).

Cette base de données est divisée en données d'entraînement, de validation et de test, et a été initialement fournie par l'utilisateur cchangcs <https://www.kaggle.com/asdasdasdasdas/garbage-classification>.

Toutes les données sont de même taille (512 x 384 pixels) et en couleurs (RGB), avec une résolution de 96 pixels par pouce. Sur les exemples d'images ci-contre, de gauche à droite : (i) une canette en métal, (ii) une bouteille en plastique et (iii) un carton. Nous avons augmenté cette base de données en y ajoutant des images modifiées (retournées, tournées, zoomées, etc.) à l'aide de la classe ImageDataGenerator de la bibliothèque.



FIGURE 3.1 – Exemple d'images de la base de données

2 Description de la méthode d'apprentissage en profondeur utilisée

2.1 Xception

Le modèle de Xception est un modèle de réseau de neurones convolutionnel (CNN) pour la reconnaissance d'images utilisant l'apprentissage en profondeur. Le modèle Xception a remporté le premier prix à l'ILSVRC (ImageNet Large Scale Visual Recognition Challenge) en 2016, battant ainsi tous les autres modèles de l'époque en termes de précision.

La principale innovation de Xception est l'utilisation de convolutions séparables en profondeur, qui permettent de réduire considérablement le nombre de paramètres du modèle tout en conservant une précision élevée. Le modèle Xception est également connu pour sa rapidité d'exécution, ce qui en fait un choix populaire pour les applications en temps réel. Nous avons utilisé ce modèle sur notre base de données avec plusieurs techniques de supervision de l'entraînement (Early stopping, learning curve...).

2.2 YOLOv3

YOLOv3 (You Only Look Once version 3) est un modèle de détection d'objets basé sur l'apprentissage en profondeur. Il est connu pour sa vitesse de détection rapide et sa précision élevée, ce qui en fait un choix populaire pour les applications de détection d'objets en temps réel.

Le modèle YOLOv3 utilise une architecture de réseau de neurones convolutionnel (CNN) en couches profondes pour prédire les classes et les coordonnées des boîtes englobantes autour des objets dans une image. Il est également capable de détecter plusieurs objets dans une seule image en utilisant un cadre de détection unique, ce qui le distingue des modèles de détection d'objets précédents qui nécessitaient un cadre de détection par objet.

3 Description du déploiement du modèle avec Docker et GitHub

Avant tout, il a été nécessaire d'installer Docker sur la machine locale de chaque membre de notre groupe. Ensuite, on a créé un nouveau dépôt sur GitHub et on l'a cloné sur la machine locale de l'un des membres du groupe. Par la suite, les autres membres du groupe peuvent ont été ajoutés comme collaborateurs au dépôt en utilisant les paramètres de collaboration de GitHub. Cela leur permettra de pusher et de pull les modifications au dépôt.

L'étape suivante a été de créer un fichier Dockerfile dans le répertoire racine du dépôt. Le Dockerfile est un fichier de configuration qui définit les instructions pour construire l'image Docker. Il peut inclure des instructions pour installer les dépendances logicielles nécessaires, copier les fichiers du modèle et de l'application dans le conteneur, et définir les commandes

à exécuter lorsque le conteneur est lancé. Par la suite, l'image Docker a été construite en utilisant la commande `docker build`. Cette commande prend en entrée le Dockerfile et génère une nouvelle image dans le référentiel local de Docker.

CHAPITRE 4

RÉSULTATS

- 1 Présentation des résultats de l'entraînement du modèle
- 2 Evaluation du modèle sur des données de test

CHAPITRE 5

DISCUSSION ET CONCLUSION

Pour résoudre les problèmes de détection et de classification des déchets dans les centres de tri, un modèle d'apprentissage en profondeur serait sûrement le plus adapté. En effet, les modèles de deep learning ont tendance à être plus robustes et à donner de meilleurs résultats dans les cas où il y a du "bruit" dans les données, comme cela peut être le cas dans les centres de tri.

Nous avons vu que l'association de deux modèles de deep learning, Xception et YOLOv3, permet d'obtenir des résultats satisfaisants en termes de précision. Cependant, pour une application concrète, il serait nécessaire de tester le modèle sur un GPU (graphics processing unit) afin de s'assurer de sa viabilité en termes de temps de calcul. Il pourrait également être utile de développer un modèle spécifiquement adapté aux bounding boxes de Xception, et d'avoir une base de données plus étendue pour entraîner et évaluer le modèle.

CHAPITRE 6

ANNEXES