

**Rapport technique n°2**

**SAE 5.01** — Développement d’une application mobile de reconnaissance d’objets avec Intelligence Artificielle

**Projet :** SneakScan — Application de reconnaissance de sneakers

**Date :** 24/10/2025

**Lieu :** IUT Technologique Saulcy à Metz

**Membres du projet :**

BRAHMIA Noureddine

RAHUEL Bastien

BOUCHATROUCH Kaiss

## **1. Introduction et contexte**

Dans le cadre du projet SneakScan, notre objectif est de développer une application mobile capable d’identifier automatiquement des modèles de sneakers (chaussures de sport) à partir d’une simple photo. Pour que cette reconnaissance fonctionne efficacement, il est indispensable de disposer d’un jeu de données d’images (appelé dataset) riche et varié, représentant plusieurs modèles, marques et angles de vue.

Afin de construire ce dataset, notre équipe a développé un script Python de collecte automatique d’images, aussi appelé scraper. Ce script permet de récupérer automatiquement sur Internet des images de sneakers provenant de sites spécialisés comme FlightClub, StockX ou Nike.

Cette approche vise à gagner du temps dans la constitution du dataset, à standardiser les images récupérées et à préparer les futures phases d’entraînement de notre application de reconnaissance.

## **2. Objectif du script**

Le but du script est de collecter automatiquement des images de chaussures à partir de sites de vente en ligne.

L’idée est simple : au lieu de télécharger manuellement des centaines de photos de sneakers, le script se charge de chercher, identifier et enregistrer ces images dans un dossier organisé.

Plus concrètement, il permet :

* de rechercher une paire donnée (par exemple : Nike Air Force 1) sur un site spécialisé ;
* de repérer les pages produits correspondantes ;
* de télécharger les images affichées pour chaque produit ;
* et de les enregistrer localement, prêtes à être utilisées dans le dataset.

Cette automatisation représente un gain de temps considérable, tout en réduisant les risques d’erreurs humaines et en assurant une meilleure organisation des données collectées. Ce script permettra aussi au lancement de l’application d’étendre le dataset grâce aux utilisateurs. Effectivement si une paire n’est pas reconnu l’utilisateur pourra importer une photo et le nom du produit ce qui va permettre de créer un dataset complet immédiatement pour cette nouvelle paire.

## **3. Technologies utilisées**

Le script a été développé en Python, un langage largement utilisé pour le traitement de données et le développement d’outils d’automatisation.

Les principales bibliothèques utilisées sont :

* Requests : pour accéder aux pages web et télécharger les images.
* BeautifulSoup : pour analyser le code HTML des sites et repérer les liens et images.
* Pillow (PIL) : pour traiter les images téléchargées (vérification de leur taille, conversion en format JPEG, etc.).
* Fake User Agent : pour simuler différents navigateurs et éviter les blocages des sites.
* Tqdm : pour afficher la progression du téléchargement.

L’ensemble fonctionne de manière automatique : le script recherche les produits, télécharge les images correspondantes et les classe dans des dossiers par modèle.

## **4. Utilité et avantages du script**

La mise en place de ce script est une étape essentielle pour la réussite du projet. En effet, un modèle de reconnaissance d’images performant repose avant tout sur la qualité et la diversité du dataset sur lequel il est entraîné.

Les principaux avantages de ce script sont :

* Gain de temps : la collecte d’images est entièrement automatisée.
* Organisation des données : chaque paire de chaussures est rangée dans un dossier dédié, ce qui facilite la gestion du dataset.
* Flexibilité : il est possible de changer de site ou de modèle simplement en modifiant quelques lignes dans le script.
* Accessibilité : contrairement à certaines API payantes ou restreintes, ce script offre un accès libre et gratuit aux images disponibles en ligne.

Nous avons choisi cette solution car de nombreuses API officielles de plateformes comme StockX ou Nike sont soit payantes, soit limitées dans leur usage, soit mal documentées.

En revanche, notre script nous permet de récupérer exactement les données dont nous avons besoin, dans un format adapté à notre projet et de manière structurée.

## **5. Limites actuelles et améliorations prévues**

À ce stade, le script fonctionne correctement, mais il reste encore à l’améliorer pour répondre pleinement aux besoins du projet.

### **Limites actuelles**

* Le script ne télécharge actuellement les images que pour un seul modèle de chaussure à la fois.
* Les photos récupérées montrent souvent un seul angle de vue, alors qu’il nous faut plusieurs perspectives (dessus, côté, semelle, etc.) pour obtenir une reconnaissance plus précise. Cette limite devra être résolu pour entraîner l’IA afin qu’elle soit performante et fiable au fur et à mesure, sur sa fonctionnalité de reconnaissance de sneakers.
* Les fonds d’images sont souvent similaires, alors que nous aurions besoin de fonds variés pour rendre la détection plus fiable dans différents contextes.

### **Prochaines améliorations**

* Étendre le script pour qu’il puisse récupérer plusieurs marques et plusieurs modèles de sneakers différents.
* Modifier le code afin qu’il télécharge plusieurs angles et arrière-plans pour une même paire de chaussures, augmentant ainsi la diversité visuelle du dataset.
* Poursuivre l’optimisation de l’organisation des images et du nommage des fichiers pour faciliter leur utilisation future.

## **6. Conclusion**

Le développement de ce script marque une étape importante dans la création de notre application SneakScan.

Il constitue le point de départ de la construction d’un dataset fiable et structuré, indispensable à la réussite du projet.

En automatisant la collecte d’images, nous avons pu poser les bases d’une méthode efficace et reproductible pour enrichir notre base de données de sneakers.

Cependant, le travail reste en cours : nous allons prochainement améliorer le script afin qu’il puisse traiter plusieurs marques et modèles simultanément, et qu’il récupère plusieurs angles et fonds d’images différents pour chaque paire.

Ces évolutions permettront de créer un dataset plus complet, plus varié et plus représentatif, garantissant ainsi une meilleure qualité de reconnaissance pour notre application mobile.

