

Compte rendu N°11 : SAE 5.01 – Développement Avancé

Sujet : Création d'une application de reconnaissance par IA d'objets du monde réel en temps réel

Thème : Matériel Scolaire (stylos, cahiers, règle, etc...)

I) Avancement (26/12/25) :

Cette semaine, nous avons continué la préparation des données pour l'entraînement du modèle, nous pensons avoir fini (à moins qu'un problème inattendu surgisse).

A cause d'un problème d'ordinateur du côté d'Alexandre, qui possédait les images que nous avions prises nous-même, nous avons préféré laisser celles-ci de côté. Ainsi, nous nous sommes essentiellement concentrés sur le recueil d'images sur Roboflow. Nous avons réussi à éliminer les images erronées pour chaque type d'objet (par exemple nous avons retrouvé un manche de marteau estampillé « pen »), et équilibré l'ensemble de ces types afin de ne pas fausser l'analyse. De plus, une correction a été apportée à la méthode de dessin des cadres de reconnaissance d'image.

II) Partie Intelligence Artificielle (Said, Thomas) :

Comme spécifié ci-dessus, cette semaine nous a servi à bien avancer sur la constitution du dataset d'entraînement du modèle TF Lite, si bien que nous pensons avoir très bientôt terminé avec cette partie. Notre dataset comprend 9 classes :

- Les stylos (« pen »)
- Les crayons (« pencil »)
- Les surligneurs (« highlighter »)
- Les taille-crayons (« sharpener »)
- Les gommes (« eraser »)
- Les règles (« ruler »)
- Les ciseaux (« scissors »)
- Les tubes de colle (« glue_stick »)
- Les agrafeuses (« stapler »)

Ainsi, ces 9 types d'objets scolaires seront reconnus par notre application. Les images erronées (par exemple un manche de marteau estampillé « pen », ou une image contenant effectivement un stylo, mais avec l'entièreté de l'image sélectionnée, alors que celle-ci contient beaucoup d'autres choses, comme une feuille, de l'écriture, etc...) ont été supprimées, et les différentes classes ont été équilibrées. Cette dernière étape est cruciale pour ne pas fausser l'analyse de l'IA, surtout pour les stylos et crayons, très semblables. La prochaine étape est donc d'entraîner notre modèle d'IA, puis de tester si l'entraînement a apporté les résultats voulus. Si c'est le cas, nous pourrons alors passer à la création des deux fonctionnalités manquantes : la reconnaissance d'objets importés depuis la galerie, et l'historique des reconnaissances d'objets.

III) Partie Développement et Architecture (Thomas) :

En dehors des aspects liés au modèle, nous avons apporté une légère amélioration à la méthode de dessin du cadre de reconnaissance d'objets de l'application, en fonctionnement bien que notre modèle d'IA n'ait pas encore été entraîné à reconnaître des objets scolaires. Voici les détails des points positifs apportés par cette amélioration :

- Si l'objet détecté se trouve en haut de l'écran, le texte est automatique déplacé à l'intérieur du cadre afin de rester visible.
- La logique mathématique d'inversion de l'image est différente, elle est plus robuste, et plus simple pour Flutter. Cette méthode évite de possibles décalages dans le cas où le widget ne serait pas parfaitement aligné sur l'écran.
- Suppression de l'argument « `widgetSize` », inutilisé ici.
- Utilisation de constantes pour le style du texte, ce qui est une meilleure pratique de performance en Flutter pour éviter de recréer l'objet à chaque rafraîchissement de l'image (environ 30 fois par seconde pour la caméra).

IV) Prochaines étapes :

- Entrainer le modèle d'IA
- Reprendre les anciens fichiers Dart, concernant l'interface utilisateur, provenant du main, ils sont plus complets et ont vu un apport de diverses améliorations
- Rendre fonctionnelle la fonctionnalité de reconnaissance d'images provenant de la galerie
- Implémenter l'historique des détections
- Améliorer les messages d'erreur pour l'utilisateur