

# Rapport de séance

Durant cette séance j'ai tenté d'utiliser l'image JetBot sur la jetson afin d'utiliser ses capacités conçues pour le suivi d'objet. Cela me paraissait être une bonne option puisque mis à part l'image à mettre sur la carte, je n'avais rien d'autres à télécharger, ce qui était la source de la plupart des problèmes que j'avais rencontré jusque-là. J'ai donc utilisé une autre carte SD pour y mettre l'image JetBot. J'ai d'abord eu besoin d'un écran afin de configurer la carte pour ensuite m'y connecter en ssh. Après cela j'ai pu me connecter à la carte via jupyter lab afin de programmer un code python.

```
from jetbot import ObjectDetector  
  
model = ObjectDetector('ssd_mobilenet_v2_coco.engine')
```

J'ai donc commencé par exécuter ces deux lignes. La première était correct, ce qui confirme que je n'ai pas besoin de télécharger un package particulier. En revanche la deuxième a eu un problème. Après plusieurs recherches, j'ai trouvé que cela était lié au fait que le modèle que je voulais utiliser (un modèle télécharger directement sur le site de NVIDIA) n'était pas compatible avec l'image JetBot que j'avais prise. J'ai donc tenté de résoudre ce problème mais sans succès.

Je me suis donc remis sur l'idée d'utiliser la bibliothèque cvzone afin de détecter une main ainsi que sa position pour la suivre. J'ai donc réinstallé l'image précédente sur ma carte SD afin d'être sûr que tous les téléchargement faits précédemment ne me rendent pas la tâche plus difficile. Après avoir de nouveau configurer la carte, j'ai créé un environnement virtuel afin de ne pas tout mélanger sur la carte, et donc pour augmenter mes chances de réussites. Dans celui-ci, j'ai donc installé python 3.8 et mis à jour la version de pip. Ensuite j'ai installé cvzone avec succès. Cette fois-ci, afin d'effectuer mon code python, j'ai décidé d'utiliser l'éditeur de texte nano, cela m'évite de devoir retélécharger cvzone dans le docker (Ce qui était un problème parmi mes manipulations précédentes). J'ai également pu télécharger certaines bibliothèques nécessaires au bon fonctionnement de différentes utilisations de cvzone.

J'ai finalement pu tester avec succès l'importation de cvzone dans un programme python sur nano. La prochaine fois, je devrais donc pouvoir commencer à programmer en python afin d'extraire les informations de positions d'une main détectée par une caméra afin de les envoyer à l'Arduino.