

Rapport : Object Detection avec MediaPipe & OpenCV

Le but du projet est de créer un système simple permettant de détecter des objets en temps réel. Pour cela on essaye différents modèles pré entraînés sur coco dataset.

Le système est facile d'utilisation, il suffit d'installer les packages et d'exécuter main.py.

Trois modèles ont été testés, `efficientdet_lite0`, `efficientdet_lite2` et `ssd_mobilenet_v2`. On mesure les performances du système en FPS avec chacun de ces modèles.

	Detection FPS	Overall FPS
<code>mobilenet_v2</code>	$\simeq 40$	$\simeq 30$
<code>efficientdet_lite0</code>	$\simeq 25$	$\simeq 21$
<code>efficientdet_lite2</code>	$\simeq 9$	$\simeq 7$

efficientdet_lite2 offre une fluidité insuffisante il n'est pas adapté à notre cas d'utilisation, **efficientdet_lite0** offre une fluidité correcte. Quant à **mobilenet_v2** la fluidité est très bonne mais en l'utilisant on remarque que le modèle n'est pas très confiant dans ses prédictions. En effet, par exemple, lorsque je me tiens en face de la webcam avec une bonne luminosité **mobilenet_v2** donne une probabilité d'environ 85% que je sois une personne, alors que **efficientdet_lite0** est plus confiant et donne une probabilité autour de 90%. **efficientdet_lite0** semble donc être le meilleur compromis entre fluidité et précision, ce qui n'est pas une surprise puisque c'est ce qui est mentionné dans la documentation de MediaPipe.

Pour tester la précision des modèles, on utilise différents objets présents dans le cocodataset (pomme, téléphone, verre, télé, canapé etc ...) avec différents niveau de luminosité. On remarque que la détection devient difficile lorsque la luminosité baisse et lorsque l'objet est en arrière-plan, mais en général elle reste correcte avec **efficientdet_lite0**.

On aurait également pu faire des essais avec le framework YOLO, connu pour offrir des algorithmes de détection en temps réel efficaces.