

# Rapport Projet Traitement de Signal

---

2021-2022

Bombaert Andréas, Brinchant Vincent, Carlier Louis, De Dryver Cédric,  
Dieuzeide Gaël, Schamroth Arthur.

EPHEC

## A) Introduction

Dans le cadre de notre cours de traitement de signal, nous avons dû élaborer un programme permettant la reconnaissance d'objets, de fruits, de signes, ou autres. Le but étant d'apprendre le machine learning.

Nous avons la possibilité si nous le souhaitons d'intégrer ce projet à notre projet d'intégration, c'est ce que nous avons décidé de réaliser dans notre cas.

Pour implémenter ce projet à notre cours d'intégration, nous avons réfléchi à une fonctionnalité qui pourrait s'intégrer parfaitement dans celui-ci sans que cela ne soit hors-contexte. Afin de mieux expliciter notre projet de signal, nous allons brièvement vous présenter notre projet d'intégration.

Celui-ci est une application citoyenne mettant en relation plusieurs individus désirant réaliser des actions écologiques dans leur ville, leur environnement. Ces actions sont diverses et variées, celles-ci peuvent aller d'un simple ramassage de déchets en plus ou moins grande quantité, nous les regroupons sous le nom de collecte, ou bien d'un scan de ticket de transport en commun. Les participants à ces différentes actions gagnant des réductions, des cadeaux via un système de points de fidélité dans l'application et grâce à des magasins partenaires.

Ainsi, nous avons décidé d'intégrer à cette application mobile une nouvelle fonctionnalité permettant à ses utilisateurs de déterminer si oui ou non un déchet avait sa place dans du compost, soit via un système de prise de photo stockée dans le téléphone, soit directement via la caméra du téléphone.

Nous avons donc plusieurs objectifs au sein de ce projet, nous souhaiterions en apprendre davantage sur le fonctionnement du machine learning et réussir à mettre en pratique nos nouvelles connaissances. Également, nous souhaiterions réussir à rassembler ce projet de machine learning dans notre projet d'application citoyenne, le fait de réunir un projet kotlin et un projet de machine learning écrit en python pouvant être compliqué à mettre en place car ce sont deux langages très différents.

## B) Solutions existantes

Une multitude de solutions existe concernant le domaine du machine learning, nous nous sommes intéressés à plusieurs d'entre elles, notamment, OpenCV, caffe, Pytorch, YoloV5, Roboflow et bien d'autres. C'est sur ces deux dernières technologies que nos choix se sont portés. En effet, YoloV5 est la version la plus récente et performante de l'environnement Yolo permettant la reconnaissance d'objet, de personnes en direct ou sur des images fixes.

Nous avons décidé d'utiliser l'outil Yolo car il s'agissait de la solution la plus proposée et conseillée lors de nos recherches, de plus de nombreux outils sont disponibles et la documentation de Yolo est extrêmement détaillée et pratique.

Yolo fait partie du projet PyTorch, qui est une bibliothèque logicielle en Python, open source, qui permet de faire de l'apprentissage machine (Machine Learning), qui s'appuie sur Torch qui est un framework développé par Facebook Research.

Andréas Bombaert, Arthur Schamroth,  
Cédric De Dryver, Gaël Dieuzeide,  
Louis Carlier, Vincent Brinchant

Parmi les outils en collaboration avec YoloV5, nous retrouvons Roboflow qui est un outil nous permettant d'utiliser des datasets directement dans nos projets. Son utilisation est quelque peu fastidieuse dans un premier temps, mais une fois la documentation lue et comprise, son utilisation est un véritable gain de temps.

En effet, Roboflow nous permet d'accéder à un vaste choix de datasets créés par la communauté et de les importer dans nos projets. Ces datasets contiennent ainsi une multitude d'images ainsi que leurs labels, c'est-à-dire l'emplacement des différents objets, personnes, éléments dans ces images.

Roboflow nous permet également, et c'est sans nul doute son plus gros atout, de créer nos propres datasets à partir de nos images. Nous devons ainsi importer dans Roboflow une grande quantité d'images contenant les éléments que nous désirons identifier dans notre dataset et les découper afin de pouvoir les isoler du reste de l'image. En réalisant ces découpages, les labels sont automatiquement créés et sont ainsi directement utilisables. En nous renseignant sur ces labels, nous n'avons trouvé nulle part une documentation nous expliquant comment créer soi-même ses propres labels dans ses datasets, sans passer par des outils tiers comme Roboflow, nous nous sommes ainsi concentrés sur cette solution.

## C) Partie Technique

Comme expliqué précédemment, nous nous sommes penchés sur la solution de YoloV5 dans notre projet, cette technologie étant simple à utiliser et bien documentée. Nous avons ainsi dû récupérer la structure d'un projet Yolo et le modifier pour qu'il soit fonctionnel avec notre projet.

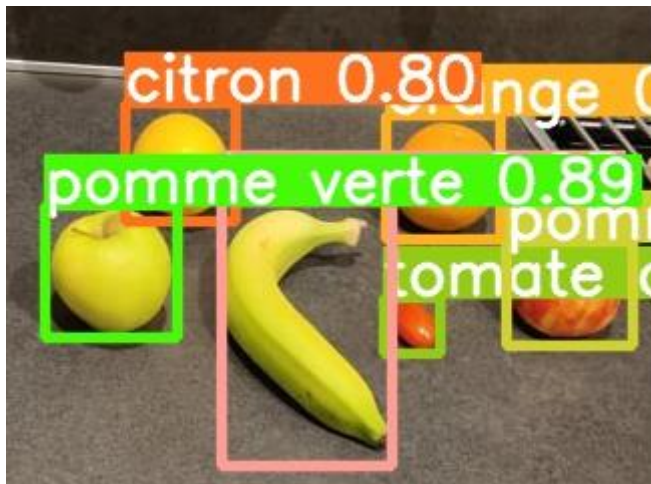
Ainsi, nous avons chacun créé un ou plusieurs datasets de fruits et/ou de déchets, nous avons « entraînés » ces différents datasets avant de les regrouper en un, afin que ceux-ci soient plus efficaces et plus rapides lors de la détection d'objets sur image fixe ou en direct via caméra.

L'un des atouts majeurs de Yolo est le fait qu'il enregistre chaque test, entraînement, dans ses fichiers, que ce soit sur des images statiques, le programme enregistre ainsi des .jpg, ou bien via la caméra, le programme enregistre ainsi des vidéos accélérées sous le format .mp4.

Ayant directement repris la structure et le fonctionnement d'un projet Yolo habituel, nous n'avons pas modifié beaucoup de points dans les algorithmes, mais nous avons énormément travaillé sur l'aspect des datasets, bien que Roboflow ait été là pour nous faciliter la tâche, il ne faut pas négliger l'importance du nombre d'images et de label dans ces datasets, sans quoi, ceux-ci n'auraient pas été suffisamment performant que pour détecter le moindre élément désiré.

## D) Résultats

Voici un exemple de résultat enregistré par notre programme :



Sur celui-ci, nous pouvons constater les points forts et les points faibles de YoloV5. Parmi ces différents points forts, nous pouvons retrouver la précision des détections des différents éléments présents sur cette image, ceux-ci sont correctement localisés et reconnus avec un taux de certitude assez élevé (tous au-dessus de 80%). C'est sans nul doute l'un des points qui nous avait le plus marqué lors de notre choix de technologie, le degré de précision de YoloV5 si et seulement si les datasets sont correctement implémentés.

Néanmoins, nous retrouvons également quelques points négatifs parmi ceux-ci. Nous pouvons parler du manque de lisibilité dans certaines situations, nous rencontrons déjà ce problème dans la capture d'écran ci-dessus, l'image est directement coupée sur la droite et nous ne pouvons lire correctement ce qu'il est indiqué, que ce soit le nom de l'objet ou son degré de précision. Bien que cela ne soit qu'un détail, cela pourrait nous handicaper si nous voulions absolument obtenir des données précises sur chacune de nos images.

La richesse des datasets est également un point très important pouvant être à la base de beaucoup d'erreurs, si ceux-ci ne sont pas suffisamment fournis en images et en labels, la détection d'éléments peut s'avérer compliquée dans certains cas. La documentation officielle de Yolo nous recommande de fournir pas loin de 1500 images et labels pour obtenir un dataset suffisamment précis en toute circonstance. Nous avons ainsi dû nous concentrer sur les fruits et déchets principaux. Si nous avions décidé de nous concentrer sur un grand nombre de fruits et de déchets, nous n'aurions sûrement pas su être aussi précis que nous ne le sommes dans la capture d'écran ci-dessus.

Pour avoir autant de fruits et de déchet, nous avons divisé la création de datasets de fruit entre les membres du groupe. Chacun à créer au moins un dataset de fruit, puis nous avons tout regrouper en un seul dataset, ce qui nous permet d'avoir beaucoup de fruits que nous pouvons détecter et indiquer à l'utilisateur lesquelles peuvent être mis dans un compost.

Ce dernier point nous porte à croire qu'il peut s'agir d'une réelle limite dans le cadre de projets plus conséquents. En effet, si dans le cadre d'un projet nous devons intégrer la reconnaissance d'un grand nombre d'éléments, nous devons intégrer à notre dataset une quantité d'images conséquente, sans quoi le dataset ne sera suffisamment efficace et fourni. Et nous serions ainsi tenter d'utiliser une autre technologie que Yolo nécessitant peut être moins d'images et de labels que ce dernier, encore faut-il qu'une telle technologie existe bien évidemment.

## E) Conclusion

Nous avons dans le cadre de ce projet, énormément appris dans le domaine du machine learning et nous avons réussi à mettre ces nouvelles connaissances en pratique. Néanmoins, nous avons remarqué que nous ne réalisons pas réellement de code nous-même, en effet, la majorité de notre programme est issue directement de la structure de base du projet Yolo, nous n'avons pas dû grandement le modifier pour qu'il soit opérationnel dans notre projet. Nous avons uniquement dû modifier le dataset utilisé et le reste du programme était prêt à faire tourner notre projet. Mais créer un dataset de plusieurs fruit n'a pas été simple. Nous avons passé beaucoup de temps à trouver une solution optimale et puis à apprendre comment maîtriser cette technologie afin d'avoir des résultats de détection satisfaisant pour ce que nous cherchons à faire, en passant par des grosses étapes de recherche et d'entraînement de notre programme pour créer un dataset intéressant.

En piste d'amélioration pour notre projet, nous pourrions ainsi continuer à enrichir notre dataset pour que celui-ci soit de plus en plus efficace et rapide. Nous ne voyons pas réellement ce que nous pourrions modifier dans le code Yolo tant celui-ci nous semble correctement implémenté. Néanmoins, nous pourrions continuer à intégrer ce projet de machine learning à notre projet d'intégration. En effet, nous souhaiterions ajouter lors de notre présentation finale du projet d'intégration la fonctionnalité de reconnaissance de déchet.

## F) Avis Personnels

### 1) Schamroth Arthur

Lors de ce projet, j'ai énormément appris concernant le machine learning, cela a été grandement bénéfique pour moi et je m'y intéresse à présent beaucoup plus qu'auparavant. Il a été très motivant de voir nos modèles de données grandir et s'améliorer au fur et à mesure des trains que nous exécutions et également de comprendre peu à peu le fonctionnement d'un projet Yolo. Néanmoins, j'ai été quelque peu étonné de voir que des solutions étaient directement à disposition sur internet, nous n'avons pas réellement dû modifier grand-chose dans le code source du projet et je m'attendais à devoir coder plus que ce que nous avons dû faire.

### 2) Carlier Louis

Ce projet fut très intéressant car il m'a permis pour la première fois d'explorer et de travailler dans le domaine du machine learning et la reconnaissance d'objets. Cependant la réalisation de ce projet n'a pas été simple, en effet nous avons passé beaucoup de temps à chercher et à expérimenter diverses solutions pour arriver à nos fins afin de trouver l'alternative la plus adaptée à notre projet d'intégration. Concernant l'organisation au sein du groupe, une fois le choix fixé de la solution que nous allions utiliser, nous nous sommes répartis de manière efficace les tâches. Comme la construction des différents datasets pour les différents déchets ainsi que l'entraînement de ceux-ci, cette partie a d'ailleurs pris beaucoup temps jusqu'à ce que nous obtenions des résultats concluants sur la reconnaissance des déchets.

### 3) Dieuzeide Gaël

L'aspect dont j'aimerais parler sont les différents outils avec lesquels nous avons avancé en groupe sur la réalisation d'un modèle.

Tout d'abord, pour le choix des technologies, chacun ayant été mis au courant des possibilités, nous avons dû choisir celle qui nous paraissait la plus appropriée sans avoir de connaissances profondes sur le sujet.

Pour la réalisation et l'entraînement de nos modèles respectifs, nous avons utilisé un google collab qui a permis à chacun de rester à niveau et d'entraîner son modèle.

Ensuite, pour mettre en commun les modèles nous avons utilisé un système d'importation proposé par roboflow et chacun a pu partager le fruit de son travail et nous avons pu tout mettre en commun.

Enfin, Pour la partie application et le code, nous avons utilisé un github directement connecté avec VCS à nos Pycharm pour partager et compléter le code tous ensemble.

### 4) Bombaert Andréas

De mon point de vue nous nous y sommes pris un peu trop tard mais nous avons travaillé ensemble pour essayer de fournir un dataset le plus complet possible pour certains fruits afin de prouver que nous pouvions fournir de la reconnaissance de déchets ménagers avec une certaine précision. Nous manquons cependant de maîtrise en ce qui concerne le code python de yoloV5 et cela nous a ralenti énormément car les documentations pour modifier le code de base de yolo sont assez rares. De plus Kivy est assez réticent par rapport à l'affichage d'images, nous mettant des batons dans les roues.

### 5) Brinchant Vincent

J'ai trouvé ce projet très intéressant car le machine-learning est une matière qui m'intriguais beaucoup avant de démarrer ce projet. J'ai beaucoup appris et j'ai vraiment apprécié voire s'améliorer mes dataset et les résultats que nous obtenions. Je trouve néanmoins dommage que la matière n'a pas été abordée plus tôt dans l'année.

Bien que nous ayons commencé le projet trop lentement en testant beaucoup de solution par manque de connaissances, nous avons réussi à nous organiser pour terminer efficacement ce projet. Nous nous sommes tous impliqué et nous avons une bonne collaboration mais je considère que j'aurais pu faire plus sur ce projet.

### 6) De Dryver Cédric

Personnellement j'ai eu beaucoup de mal à comprendre comment fonctionnait le machine learning, et comment s'utilisait l'outil. Mais grâce aux ressources que le groupe a pu mettre en place, j'ai rapidement compris comment créer mon dataset, et j'ai trouvé que c'était fort agréable que le programme puisse reconnaître le fruit que j'avais énormément photographié et testé. Le concept de machine learning me faisait assez peur au début car je pensais que c'était très compliqué à manipuler, mais finalement ce n'était pas si difficile et j'ai pu apprendre sans trop de difficultés. Pour conclure, utiliser du machine learning dans le cadre de ce projet m'a donné des idées intéressantes pour des projets à venir et, de plus, pouvoir partiellement maîtriser cette compétence est un bon point pour le futur.

## G) Sources

- Kivy: <https://kivy.org/doc/stable/api-kivy.app.html>
- YoloV5 sur Youtube:
  - [https://www.youtube.com/watch?v=GRtgLlwxpc4&list=LL&index=5&ab\\_channel=DeepLearning](https://www.youtube.com/watch?v=GRtgLlwxpc4&list=LL&index=5&ab_channel=DeepLearning)
  - [https://www.youtube.com/watch?v=h-PlsIPNael&list=LL&index=4&t=701s&ab\\_channel=PeiYang](https://www.youtube.com/watch?v=h-PlsIPNael&list=LL&index=4&t=701s&ab_channel=PeiYang)
  - [https://www.youtube.com/watch?v=tFNJGim3FXw&list=LL&index=3&t=4133s&ab\\_channel=NicholasRenotte](https://www.youtube.com/watch?v=tFNJGim3FXw&list=LL&index=3&t=4133s&ab_channel=NicholasRenotte)
  - <https://www.youtube.com/watch?v=MdF6x6ZmLAY&t=227s>
  - <https://www.youtube.com/watch?v=cXDzzefyGPs>
  - <https://www.youtube.com/watch?v=oQ0436IJUWc>
  - [https://www.youtube.com/watch?v=YIRd4rw\\_vBw](https://www.youtube.com/watch?v=YIRd4rw_vBw)
- Github YoloV5: <https://github.com/ultralytics/yolov5>
- Google Colab YoloV5:
  - <https://colab.research.google.com/github/ultralytics/yolov5/blob/master/tutorial.ipynb>
  - <https://colab.research.google.com/github/roboflow-ai/yolov5-custom-training-tutorial/blob/main/yolov5-custom-training.ipynb#scrollTo=7iiObB2WCMh6>
- Application Python sur Mobile Youtube : <https://www.youtube.com/watch?v=ykCUSLV0pdo>
- Roboflow et YoloV5 Youtube: <https://www.youtube.com/watch?v=MdF6x6ZmLAY&t=227s>
- Roboflow Youtube: <https://www.youtube.com/watch?v=VDqsK3FDIsQ&t=522s>
- Yolo: <https://pjreddie.com/darknet/yolo/>