

Warships detector :

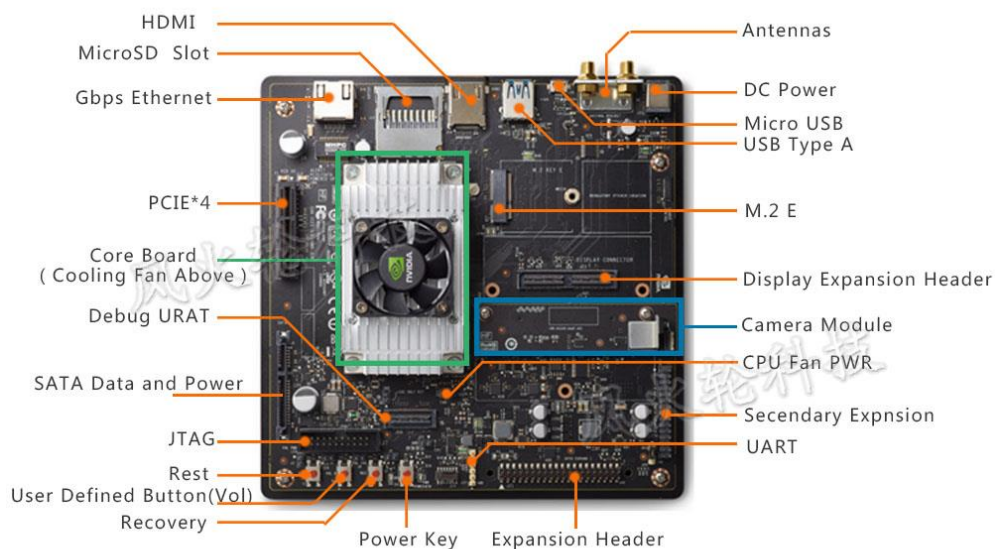
Introduction :

La gestion du trafic maritime est au cœur des préoccupations de nos armées. De plus le retour de la guerre sur le sol européen nous a montré l'importance de collecter des renseignements fiables dans un laps de temps très court c'est en considérant ces préoccupations que j'ai décidé d'orienter mon projet de traitement d'image vers la classification des navires de combat, j'ai donc décidé d'implémenter sur (nom mc) un algorithme supervisé de classification basé sur le réseau SSD (Single Shot Detector). Cet algorithme sera entraîné pour prendre en entrée une image satellite et renvoyer une classification des navires présents sur l'image d'entrée. Un objectif secondaire du projet consistera à récupérer le flux vidéo d'une caméra connectée au microcontrôleur.

Matériels :

- Cible embarquée :

La recherche documentaire sur le réseau SSD montre qu'un tel processus doit être exécuté sur un GPU. Ainsi il est envisagé d'implémenter mon algorithme sur une Jetson Xavier TX2. Ainsi le dataset de en Pascal-VOC et donc le SSD pourra être implémenter sur jetson TX2.

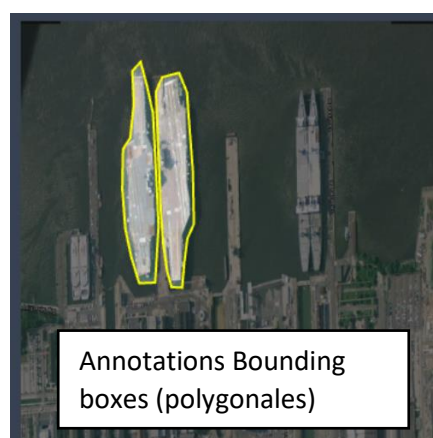
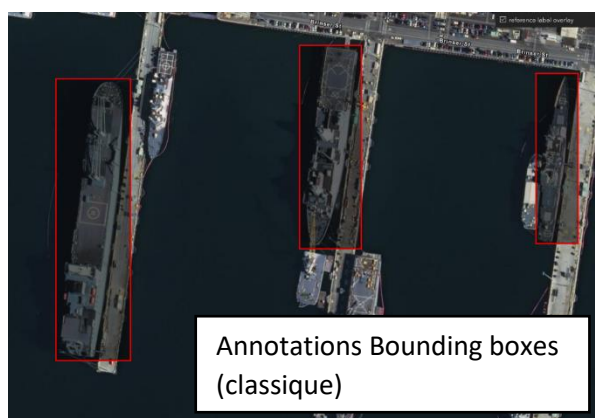


Pour ce qui est du réseau YOLOV5 nous pourrions le faire fonctionner sur un Raspberry pi 4 munit d'une version 64 bits d'Ubuntu. (Cette implémentation sur Yolo via un Raspberry a été expérimenté suite a des complications avec SSD. Vous retrouverez donc les deux démarches).

- Dataset :

Ce projet visant à détecter les navires de combats présents sur des images satellites se base sur un jeu de données composés de 278 images, issue de la page [World imagery Wayback](#) qui permet de visualiser toutes les images prises pour une position géographique donnée durant les dix dernières années. Ceci m'a permis de récupérer un grand nombre d'images afin d'alimenter mon jeu de données.

Une fois les images sélectionnées il faut les annoter pour ce faire il existe plusieurs solutions proposées sur internet. Durant ce projet nous en avons testé deux la première qui consiste à des annotations via bounding boxes sur la plateforme [CHOOCH AI](#) et la deuxième via bounding boxes polygonales sur [Roboflow](#) permettant une meilleure sélection de la région à annoter.

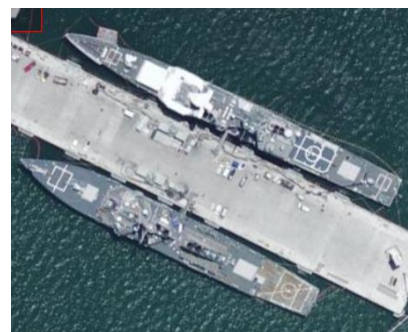


Le format de ces données est le suivant :

- L'image de base au format jpg.
- Les annotations au format Xml une par image (pour les jeux de données type Pascal-Voc).
- Un fichier d'annotation au format Xml (pour les jeux de données type COCO-Json).

Nous souhaitons entraîner le réseau YOLOv5s pour détecter les bateaux ainsi, nous utiliserons un jeu de données issue de Roboflow au format YOLOv5. Le jeu de données de CHOOCH AI lui a été utilisé afin de mener un essai d'entraînement d'un réseau type SSD (single shot detector). Celui-ci est malheureusement resté sans succès. Le format et l'utilisation du jeu de données ayant été affecté, nous devons maintenant préciser la composition de celle-ci ainsi que ses particularités. Ce jeu de données comporte 11 classes correspondantes chacune à un type de navire de combat différent. Le tableau en annexe vous donne le listing des classes ainsi que le nombre d'images et d'itérations présentés dans le jeu de données. Même si les différences entre les classes semblent importantes pour l'œil humain, il est fort à parier que le détecteur rencontrera des difficultés à prédire la classe exacte de certains navires pour la simple et bonne raison, j'ai rencontré les mêmes difficultés lors de l'annotation. Prenons un exemple très parlant :

Sur cette image, on peut retrouver deux classes différentes. On





peut voir qu'il y a des différences entre ces navires en termes de profil de coque avec pour le navire 1 une forme plus longiligne que pour le navire 2. Cependant l'aspect deux dimensions des Image satellite ne donnant pas de profondeur et la prise de vue par le dessus ne permettant pas d'avoir accès aux informations visuelles classiquement utilisé pour différencier deux navires (Type de radar,



Il est bien plus facile de les distinguer sous cet angle, on peut voir que la mature n'est pas du tout la même (cerclée en rouge). Tout cela pour dire que la prise de vue d'un satellite ne permet que de discriminé les bateaux selon deux critères la taille, la forme globale et l'aspect général des matériels installé sur celui-ci. De ce fait, on peut supposer que le dataset avec l'annotation polygonale sera plus propice, car les bounding boxes permettent de mieux épouser la forme générale du navire.

Classes :

<p><u>Carrier :</u> Navire permettant la projection de la puissance aéronavale américaine.</p> <p><u>Images :</u> 40 <u>Itérations :</u> 66</p>	
<p><u>Amphiby :</u> Navire permettant le transport de troupes et de blindés ainsi que leur débarquement</p> <p><u>Images :</u> 76 <u>Itérations :</u> 130</p>	

<p><u>Destroyer :</u></p> <p>Navire de combat remplissant tout type de mission (renseignement, lutte anti-sous-marine, défense du groupe aéronaval, etc.)</p> <p><u>Images :</u> 60</p> <p><u>Itérations :</u> 125</p>	
<p><u>Cruiser :</u></p> <p>Ce bâtiment de surface remplissant un éventail de missions très large allant de la lutte contre les narcotrafiquants à la protection du groupe aéronaval.</p> <p><u>Images :</u> 47</p> <p><u>Itérations :</u> 96</p>	
<p><u>Fregate :</u></p> <p>Bâtiment de surface remplissant des missions de défense aérienne, de projection de puissance, ainsi que de patrouille maritime et de sécurisation des voies maritimes.</p> <p><u>Images :</u> 97</p> <p><u>Itérations :</u> 315</p>	
<p><u>Pha (Porte hélicoptère amphibie) :</u></p> <p>Ce navire permet la projection des hélicoptères américains ainsi que de ses avions à décollage vertical F-35B. De plus, il possède la capacité d'emporter et de débarquer des aéroglisseurs lui permettant de projeter des troupes terrestres et de mener des opérations de débarquement aéromaritimes.</p> <p><u>Images :</u> 65</p> <p><u>Itérations :</u> 89</p>	
<p><u>Submarine :</u></p> <p>Les sous-marins sont d'une importance stratégique, car ils sont une composante fondamentale de la dissuasion nucléaire américaine. On compte deux types de sous-marins les SNLE(sous-marin nucléaire lanceur d'engins) pouvant déclencher le feu nucléaire et les SNA (sous-marin nucléaire d'attaque) ne disposant pas de l'arme atomique.</p> <p><u>Images :</u> 67</p> <p><u>Itérations :</u> 203</p>	

<p><u>Barge :</u> Bâtiment à fond plat permettant le débarquement d'homme et de matériel sur les plages.</p> <p><u>Images :</u> 11 <u>Itérations :</u> 66</p>	
<p><u>Supply :</u> Ces navires sont des bâtiments de soutien pouvant embarquer du matériel.</p> <p><u>Images :</u> 47 <u>Itérations :</u> 51</p>	
<p><u>Tender :</u> Ces navires permettent de ravitailler les autres bâtiments en vivres, matériel, carburant et tout autre type de ressources nécessaire au maintien en condition opérationnel d'un bâtiment. Il a la capacité de ravitailler en pleine mer et en mouvement les autres navires.</p> <p><u>Images :</u> 54 <u>Itérations :</u> 81</p>	
<p><u>Littoral Combat Ship :</u> Ce bâtiment furtif a pour mission de garantir la protection des ZEE (zones économique exclusive) et donc de protéger le littoral.</p> <p><u>Images :</u> 28 <u>Itérations :</u> 67</p>	

Résultats :

- Technique avec le réseau SSD :

Pour le réseau SSD un [Github](#) est disponible avec tous les codes et les résultats ainsi que les problèmes rencontrés.

- Technique utilisant Yolo sur roboflow :

Après avoir annoté les images sur Roboflow j'ai décidé de mener un entraînement directement sur la plateforme. Ensuite, pour mettre le modèle sur une Jetson ou sur un Raspberry il suffira d'utiliser l'API Roboflow directement. Pour mettre en place le modèle sur la carte, vous trouverez tout le nécessaire sur le Github suivant ainsi que le dataset. Pour exécuter le programme sur Raspberry pi 4 clonez ce [git](#) et suivez le readme du dossier Yolo. Dans la suite de ce paragraphe, vous trouverez les différents résultats des tests menés sur la plateforme Roboflow.

Equilibrage des classes :

Class Balance



Ce graphe permet de voir la répartition des classes et l'équilibrage entre elles. On peut voir que la classe frégate est sûre représentée avec 340 occurrences. Les classes destroyer, LCS, Cruiser, Barge, Supply et Tender sont-elles largement sous représentées. Ainsi, de ce graphe, on peut supposer que la détection des classes sous représentées ne sera pas très bonne avec une précision moyenne plus faible. La classe frégate qui est surreprésentée elle devrait avoir de bons scores voir même beaucoup de faux positifs dû à un surapprentissage possible.

Résultats de l'entraînement :

Training Results

chips-znaxf/3	57.6%	53.2%	58.7%
	mAP	precision	recall

Ce modèle possède un taux de rappel de 58,7 % une précision de 53,2 % et une précision moyenne 57,6 %. L'entraînement a été mené sur 300 epochs avec des Batch de 10. Je suis plutôt satisfait de ce score au vu du peu d'images utilisé et de la complexité du dataset avec des classes qui sont très proches les unes des autres (Cf exemple p2). Voyons maintenant les scores obtenus lors de la phase de test, ainsi que les prédictions faites par le modèle.

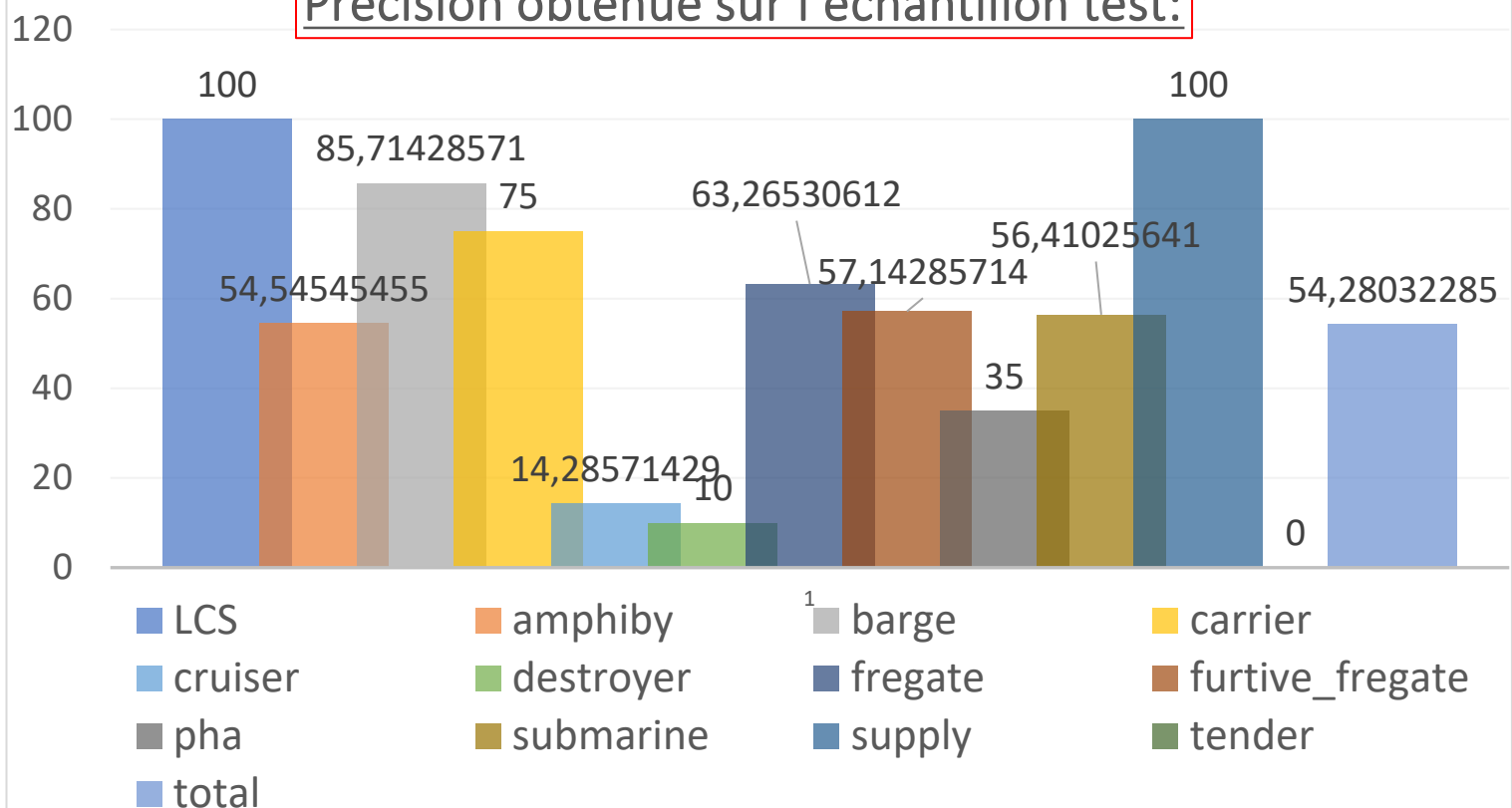
Précision moyenne par classe pour la phase de test :

Average Precision by Class



Les résultats de la phase de test ci-dessus sont obtenus sur un échantillon de 33 images. On peut voir que la mAP est de 55 % ce qui est une valeur qui est assez cohérente. Je vais plutôt attirer votre attention sur les classes LCS, Supply et furtive fregate qui ont respectivement un score de 100 %, 100 % et 90 %. Pourquoi ces valeurs sont-elles importantes. Tout d'abord, on pourrait les juger aberrantes dans un premier temps et ne pas en tenir compte. Cependant, si on regarde le critère choisi, on voit que c'est la précision qui est évaluée. Ainsi en reprenant la définition de la précision cela voudrait dire que sur 10 frégate furtive présente elle en a classé correctement 9, ou encore que sur X LCS elle en a classé correctement X. Afin de rendre ces résultats plus représentatifs de la réalité, je vous propose de regarder le graphe ci-dessous. Je l'ai obtenu en rassemblant dans un tableau un Excel le nombre d'occurrence totale (nombre de bateaux de la classe présent dans l'image + nombre d'objets classé en tant que tel.) , le nombre de faux classement (fausse prédiction + bateaux de la classe présente mais non reconnue par l'algorithme) et enfin les vrais classements correspondant aux bonnes prédictions.

Précision obtenue sur l'échantillon test:



La comparaison de ces deux graphes nous montre dans un premier temps que le pourcentage total est presque le même ce qui montre que le modèle et où la méthode d'obtention de ce graphe n'est pas totalement erronée et que si les résultats pour chaque classe diffèrent un peu, la tendance générale reste la bonne. En ce qui concerne nos 3 classes ayant une précision de plus de 90 %. Les deux premières restent avec score de 100 %, qui est à nuancer, car elles n'apparaissent qu'une fois et ont été bien classées. Le résultat qui va nous intéressait est celui des fregate_furtive. En effet, on peut remarquer de 90 %, nous passons à 57 % à contrario la classe barge passe d'un score de 32 % à 86 %. Si nous regardons maintenant ce que nous dit ce graphique, on peut voir que les classes barges, cruiser, LCS et destroyer qui sont sous représentés obtiennent de bons scores hormis pour les cruisers qui avec les destroyers sont en dessous de 20 %. Ce résultat confirme donc la difficulté que le modèle a à discriminer ces deux classes. Une solution peut être d'augmenter le nombre d'images contenant des destroyers et des cruisers afin d'obtenir de bons résultats (similaire à la classe fregate qui est-elle surreprésentée, mais qui obtient un score de 63 % alors même que les classes fregate, cruiser et destroyer sont très proches).

Voyons maintenant quelque exemple d'image sur le quel on a testé notre modèle et ou l'on peut constater des caractéristiques intéressantes :

Image 1 :

Sur cette première image on peut voir en bas à droite que le navire 1 est classé comme un croiseur et un destroyer. Ceci est dû à la proximité entre les deux classes et confirme bien notre théorie. En 2 on peut voir un que la proximité entre les différents navires semble avoir créé un faux positif pour la classe frégate. On peut cependant noter que tous les bateaux ont été détectés même si l'attribution des classes n'a pas été parfaite. Une dernière chose intéressante est la classification des deux bateaux en haut de l'image dans deux classes différentes avec une bonne prédiction sur deux. Ceci montre bien que les différences étant minimes le jeu de données est trop petit pour réussir à avoir de très bons résultats. Passons à l'image suivante.

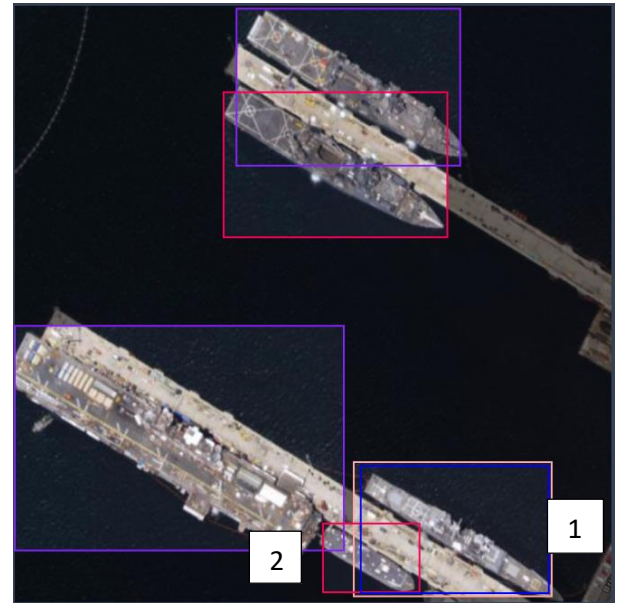


Image 2 :

Dans cette image, on peut voir en 1 la classification d'une écriture du menu comme un sous-marin, ce qui n'est pas vraiment surprenant, car la plupart des sous-marins présents dans le dataset sont une forme allongée noire avec quelques détails blancs sur un fond très foncé ainsi cette classification n'est pas surprenante. La classification en 2 elle l'est plus. En effet le navire identifié comme une frégate est en fait un destroyer furtif de classe la Zumwalt. La taille étant assez ressemblante on peut supposer que cela a suffi pour classer le bateau comme frégate.



Conclusion :

Pour conclure ce projet à pu mettre en lumière les difficultés inhérentes à la création d'un jeu de données suffisamment cohérent pour entraîner un modèle. On a pu voir sur cet exemple que les bateaux de surfaces ayant des formes très proches il sera nécessaire d'augmenter le nombre de données afin de garantir un pourcentage de détection plus élevé. Enfin sur l'utilité d'une telle application. On peut imaginer par la suite utiliser un algorithme récupérant toutes les images satellites prises pour une période donnée afin de récupérer en sortie une description de la situation tactique navale d'une puissance. Par exemple si on a tous les jours les images des bases navales d'un pays X on peut ainsi connaître la situation de ses forces et le nombre de bateaux déployés en mission le nombre qui est à quai

et les mouvements de sa flotte. Pour rendre ce système pleinement utilisable il faudra lui rajouter un programme permettant de collecter les images satellite de lui-même, et un lui permettant de synthétiser les informations et de donner un compte rendu par locations choisi.