

Détection d'objets dans des séquences d'images couleurs : compte rendu 6

Marie Bocquelet,
Fabien Caballero

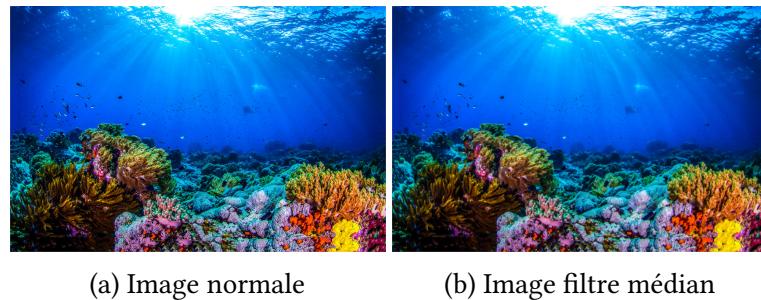
30 Novembre 2023

I Ce qui est fait :

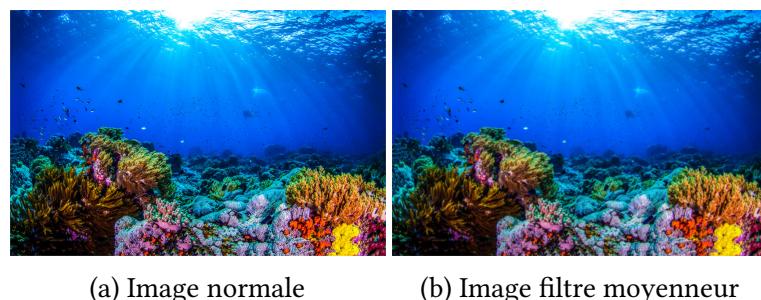
I.1 Les filtres :

Nous avons mis en place 4 filtres :

-Filtre médian :



-Filtre moyenneur :



-Égalisation d'histogramme :



(a) Image normale

(b) Image égalisée

-Égalisation de lumière :



(a) Image normale

(b) Image égalisée

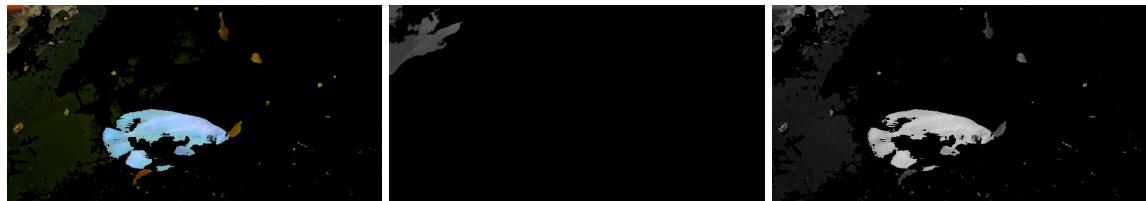
I.2 Détection et suivis par méthode traditionnelle :

Nous avons implémenté une méthode traditionnelle pour détecter les poissons. C'est une méthode par soustraction. Quelles en sont les étapes ?

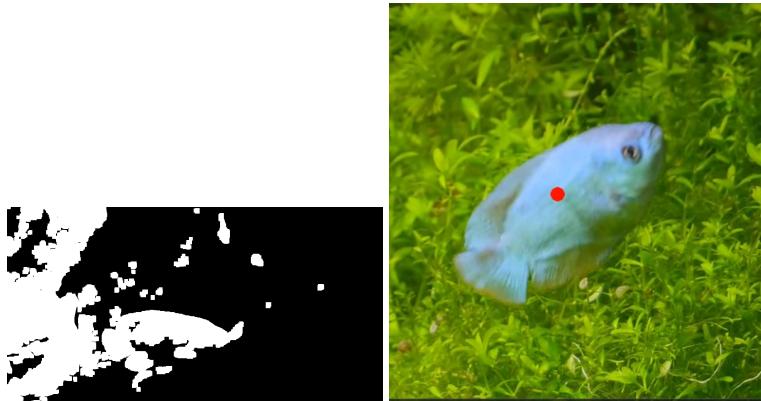
- Étape 1 : calculer le fond de la vidéo. Pour ce faire, nous faisons la moyenne de toutes les images de la vidéo.
- Étape 2 : détecter et supprimer la végétation dans le fond. Pour ce faire, nous nous basons sur les textures et les couleurs.
- Étape 3 : faire l'étape 2 pour une image de la vidéo.
- Étape 4 : convertir le fond et l'image en nuances de gris.
- Étape 5 : soustraire le fond et l'image et comparer les résultats à un seuil. Si le pixel est au-dessus du seuil, il sera blanc, sinon il sera noir.
- Étape 6 : dessiner un rond sur les pixels blancs ou les contours du masque obtenu directement.



(a) Étape 1 : calcul du fond de la (b) Étape 2 : suppression végéta-(c) Étape 3 : une image de la vidéo
tion



(a) Étape 3 : image sans végéta-(b) Étape 4 : fond en nuances de (c) Étape 4 : image en nuances
tion gris



(a) Étape 5 : masque résultat (b) Étape 6 : localisation du poisson
après seuillage

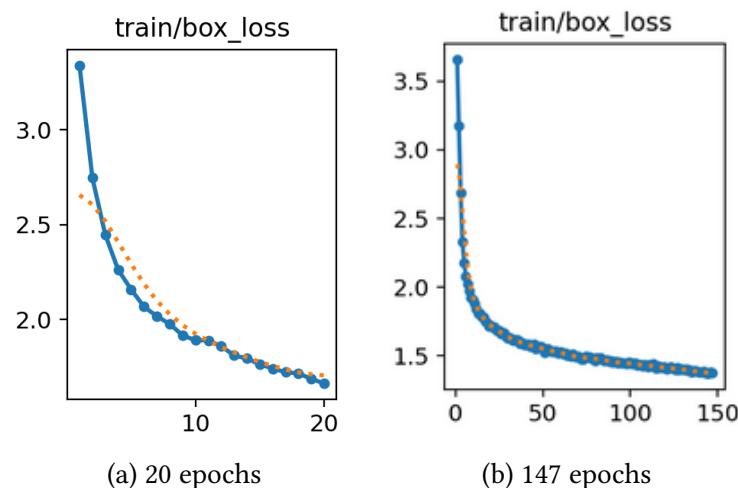
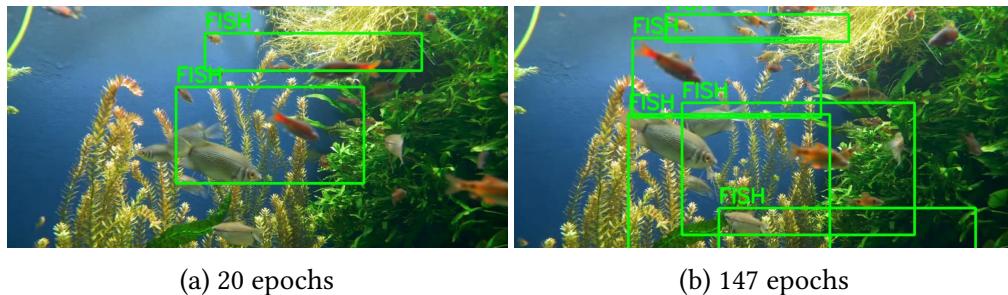
I.3 Détection et suivis par deep-learning :

Comme nous avions une obligation de résultats rapidement, nous avons décidé de d'abord nous concentrer sur un dataset obtenu sur internet : **Open Image Dataset V7**. Ce site permet de télécharger un certain nombre d'images d'un certaine catégorie, ainsi que leurs annotations déjà disponibles au format YOLO. Nous avons donc pu récupérer 6000 images de poissons et leurs annotations correspondantes.

Une fois ce dataset récupéré, nous avons entraîné un modèle YOLO. Nous avons décidé de travailler avec YOLO version 8 nano, car il est plus léger. Pour entraîner le modèle, nous avons installé un package Python appelé **Ultralytics**. Ce package permet d'entraîner simplement un modèle YOLO. Pour ça nous avons tout simplement besoin d'un fichier **config.yaml** qui contiendra les chemins

vers les images de train et de validation. Ensuite, il suffit d'utiliser une fonction train en donnant en paramètre ce fichier et les autres paramètres correspondent au batch size, nombre d'epochs...etc.

Nous avons entraîné plusieurs modèles en changeant les paramètres. Voici les résultats pour 20 et 147 epochs :



Nous avons également pu tester ces modèles sur les vidéos fournies par Quantacell :

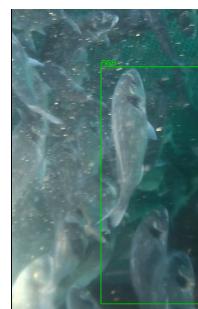


Figure 10: 147 epochs

I.4 Application :

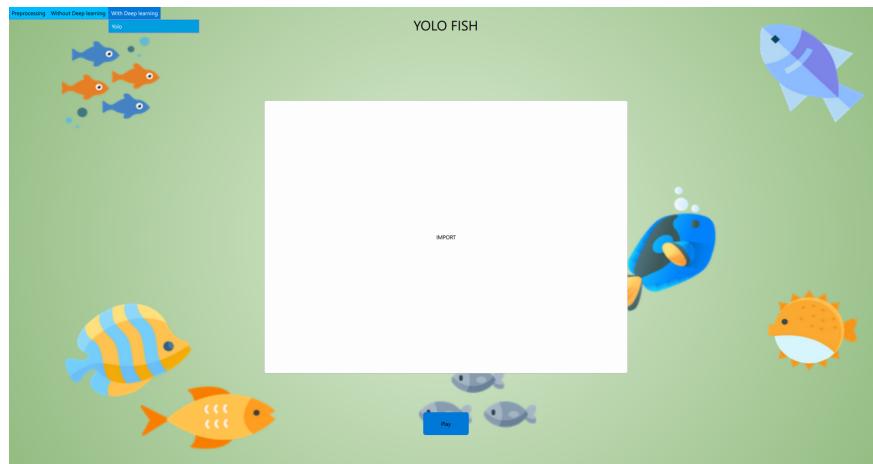


Figure 11: Apparence de l'application

L'application s'appelle **YOLO fish**. Elle permet :

-Importer une vidéo et faire play/pause

-Appliquer des filtres à des vidéos : Plusieurs filtres sont disponibles pour tenter d'améliorer la qualité de certaines vidéos sous-marines.

-Détection de poissons dans des vidéos sous-marines sans deep-learning : Nous avons mis à disposition une option permettant de détecter des poissons sans utiliser le deep-learning. Cette option demandera à l'utilisateur d'entrer un seuil souhaité.

-Détection de poissons avec deep-learning : Cette option est basée sur YOLO et permettra à l'utilisateur, soit d'entraîner son propre modèle, mais cela implique qu'il fournisse les images et leurs annotations. Soit, il pourra choisir parmi plusieurs modèles pré-entraînés sur des images de poissons.

II Ce qui n'est pas fait :

II.1 Conversion de JSON au format YOLO :

Malheureusement, nous n'avons plus le temps de trouver une solution pour la conversion du format JSON au format YOLO, bien que quelques recherches aient été effectuées.

II.2 Entraînement d'un modèle uniquement sur les données quantacell :

Comme nous n'avons pas pu avoir les annotations des données de Quantacell au format YOLO, nous n'avons pas pu entraîner un modèle uniquement basé sur les données fournies par Quantacell.

III Ce qu'il reste à faire dans le temps restant :

III.1 Terminer le poster :

Pour ce projet, nous avons dû faire un poster retracant l'ensemble des étapes du projet. Celui-ci n'est pas encore fini, mais en voici un premier aperçu :

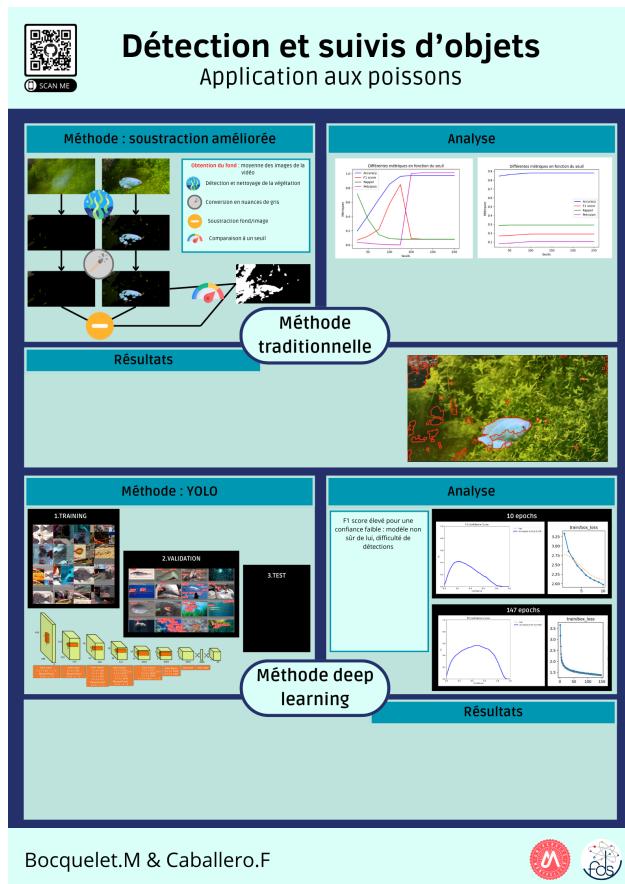


Figure 12: Poster

III.2 Préparer la présentation finale :

Nous devons encore préparer une présentation pour le 13 décembre qui retracera tout notre projet.

III.3 Tester un nouveau modèle avec un batch size différent :

Un training est en cours avec un batch size de 64 au lieu de 16 pour voir si il y a une différence.

IV Liens :

- <https://github.com/fcaball/Detection-Poissons>
- <https://trello.com/b/P8TKTbEG/projet-image>

V Citations :

Pour l'état de l'art nous nous sommes inspirés de l'article suivant : "Contribution à la détection et à la reconnaissance d'objet dans les images", de Hedi Harzallah.

Pour le deep learning nous avons utilisé le tutoriel de : <https://www.youtube.com/watch?v=Z-65nqxUd14&list=LL&index=3&t=3451s> et du github associé <https://github.com/computervisioneng/yolov8-full-tutorial>

Pour la méthode par soustraction nous nous sommes aidés de : <https://www.youtube.com/watch?v=pkzT9M1ICPE&list=LL&index=2&t=195s> et du github associé <https://github.com/L42Project/Tutoriels/tree/master/OpenCV/tutoriel13>