



INF6804 – Vision par ordinateur

Ecole Polytechnique de Montréal

Rapport de laboratoire du TP 2 - H21

élaboré par

Grégoire Chapeaux – 2033122

Mohamed Marwen Moslah - 2043911

22 Mars 2021

TABLES DES MATIÈRES

Introduction

I. Présentation des deux approches utilisées

- 1.1 Méthode de soustraction d'arrière-plan
- 1.2 YOLO (You Only Look Once)

II. Hypothèses de performance de ces deux approches dans des cas d'utilisation spécifiques

- 2.1 Mouvement d'objets avec arrière-plan dynamique
- 2.2 Mouvement d'objets avec des ombres à prédominance dure, douce et intermittente
- 2.3 Mouvement d'objets intermittents

III. Description des expériences, de la base de données utilisée et des critères d'évaluation

- 3.1 Description des expériences
- 3.2 Présentation des données
- 3.3 Métrique d'évaluation

IV. Description de l'implémentation des deux approches utilisées

- 4.1 Méthode de soustraction d'arrière-plan
- 4.2 YOLO (You Only Look Once)
- 4.3 Métrique Global-IoU

V. Présentation des résultats de tests

- 5.1 Résultats de détection
- 5.2 Mesure des Global-IoU

VI. Discussion des résultats et retour sur les hypothèses

- 6.1 Mouvement d'objets avec arrière-plan dynamique
- 6.2 Mouvement d'objets avec des ombres à prédominance dure, douce et intermittente
- 6.3 Mouvement d'objets intermittents

Conclusion

Bibliographie et références

INTRODUCTION

De nos jours, l'extraction des régions d'intérêt dans une séquence vidéo fait l'objet d'activités de recherche liées à différentes problématiques, notamment le suivi d'objets, la classification et même la régression pour des applications diverses comme la vidéo-surveillance, la vision-robotique et l'imagerie médicale.

Dans ce cadre, deux approches ont gagné de la notoriété: l'approche de soustraction avant-plan/arrière-plan (*background subtraction*), et celle de détection par classification. D'une part, la première méthode est une technique qui consiste à soustraire à l'image (*frame*) courante de la vidéo une image référence, l'image "arrière-plan". Cette méthode génère un arrière-plan, à partir des premières images en nuance de gris sur un intervalle d'acquisition suffisant. D'autre part, mis-à-part que l'approche par classification ajoute la couche identification et classification à celle de la détection d'objet(s), on ne peut pas définir d'une manière générale cette technique vu qu'elle contient plusieurs variantes qui fonctionnent avec des mécanismes assez différents les uns des autres. Dans ce TP, on choisit YOLO (You Only Look Once) comme variante de la deuxième approche qui sera étudiée d'une manière assez consistante en la comparant avec GMM (Gaussian Mixture Model), qui est une variante de la première approche.

La comparaison des deux méthodes étant particulièrement complexe, et comme cette comparaison fait l'objet d'un sujet actif en recherche représentant un vrai défi, ce TP inclut la détermination d'une métrique ayant un minimum d'objectivité, et capable de nous mener à faire une bonne analyse des résultats pour comparer la performance de ces deux approches.

I. PRÉSENTATION DES DEUX APPROCHES UTILISÉES

1.1 Méthode de soustraction d'arrière-plan

Il existe plusieurs variantes de la soustraction d'arrière-plan. Cette méthode consiste à détecter dans une image quels pixels correspondent à l'arrière-plan, et lesquels correspondent à l'avant-plan.

Nous avons choisi d'adopter une approche GMM (*Gaussian Mixture Model*) : en nous basant sur les premières frames (dans notre cas, les 50 premières images), on calcule la moyenne et la variance des couleurs de chaque pixel. On suppose que les variations des pixels suivent une distribution Gaussienne ; on estime alors, pour une frame donnée, qu'un pixel appartient à l'avant-plan si sa variation par rapport à la moyenne est supérieure à k fois la variance, avec k un facteur de sensibilité (que l'on fixe de l'ordre de 10-15). En appliquant cette méthode à tous les pixels d'une frame, on identifie donc l'avant plan et son évolution.

1.2 YOLO (You Only Look Once)

En explorant l'état de l'art [1], [2] et [3], nous avons remarqué que YOLO commençait à gagner de la notoriété durant les années récentes. YOLO fait partie des sujets centraux en traitement d'image, vision par ordinateur et apprentissage profond, et il a même évolué en plusieurs versions de YOLO-v1 jusqu'à YOLO-v4. Vu les besoins non gourmands en ressources de traitement des données, nous avons décidé de nous contenter de la version YOLO-v3 dont l'implémentation sera détaillée plus tard. L'algorithme Yolo est en particulier détaillé dans l'article "You Only Look Once : Unified, Real-Time Object Detection", publié par l'Université de Washington et AI Facebook Research Institute [1].

L'algorithme de YOLO consiste à appliquer un réseau de neurones (par exemple un réseau convolutif) en considérant l'image entièrement. Ce dernier divise l'image en une grille et détermine les boîtes englobantes (bounding boxes) qui sont déterminées via leur probabilités associées, par la méthode « régression logistique ». Ces probabilités seront seuillées via ce qu'on appelle un score de confiance qui va déterminer quelles boîtes englobantes vont réellement être utilisées pour détecter potentiellement un objet dans cette image. Cependant, il est important de discerner la différence entre le fonctionnement d'un réseau convolutif « classique » comme

CNN (Convolutional Neural Network) et celui de YOLO. En effet, CNN utilise plusieurs réseaux de neurones afin de traiter chaque « feature map » (élément représentatif de l'image) dans le but d'extraire des informations facilitant la détection d'objets comme la couleur, la luminosité, les bords orientés... En revanche, YOLO utilise ces « features maps » de la totalité de l'image pour générer les boîtes englobantes simultanément et ainsi gagner une certaine rapidité dans le processus de détection d'objet dans l'image adéquate. Ceci est illustré par la figure 1.2.1 [1] suivante :

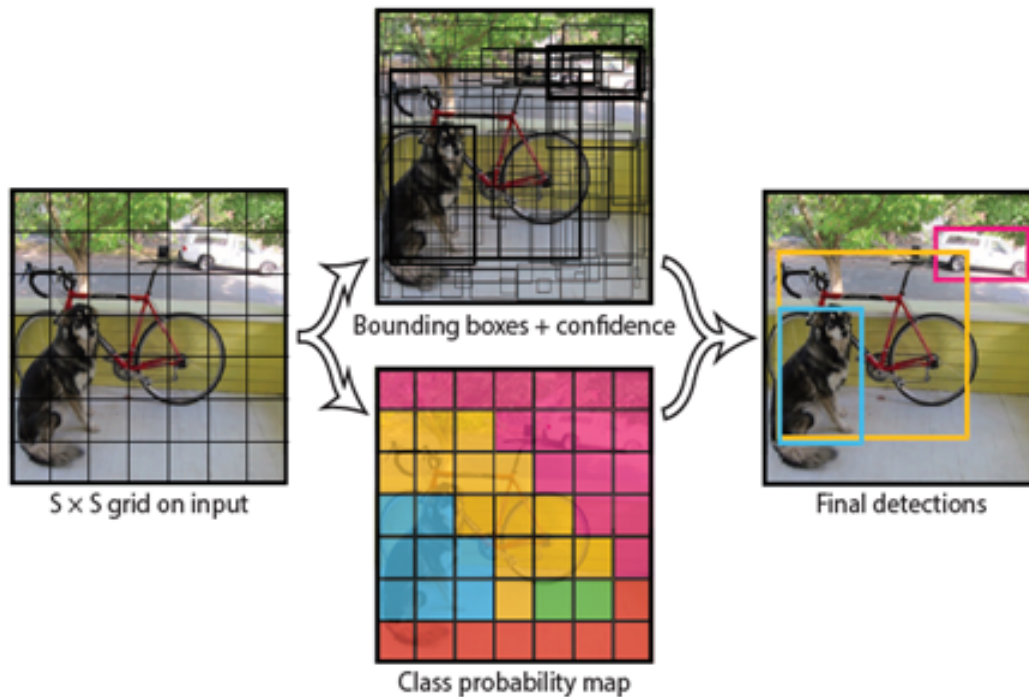


Figure 1.2.1. Une illustration simplifiée du pipeline de détecteurs d'objets YOLO

Source : [1]

II. HYPOTHÈSES DE PERFORMANCE DE CES DEUX APPROCHES DANS DES CAS D'UTILISATION SPÉCIFIQUES

2.1 Mouvement d'objets avec arrière-plan dynamique

La première hypothèse se place dans un contexte où l'arrière-plan va varier au cours du temps: par exemple, un ciel nuageux, un arrière-plan d'eau, ou un arrière-plan urbain en activité. Comme la méthode de soustraction d'arrière-plan repose sur la génération d'un arrière-plan à partir de la moyenne et de la variance des frames, on pourrait supposer qu'un arrière-plan dynamique sera plus difficilement reconnaissable par cette méthode.

Cette intuition est déjà confirmée dans l'article [4] par la phrase suivante: "However, the [GMM] algorithm does not provide appropriate results in case of radical changes [...] or **movements in the background**".

On va donc formuler et chercher à vérifier l'hypothèse suivante : **dans un contexte d'arrière-plan dynamique, un détecteur par classification tel que Yolo sera bien plus performant qu'une méthode de soustraction d'arrière-plan.**

2.2 Mouvement d'objets avec des ombres à prédominance dure, douce et intermittente

Dans cette deuxième hypothèse, on se place dans un contexte où les objets ont des ombres avec une prédominance variée. Dans cette perspective plusieurs cas d'utilisations pourraient avoir lieu :

- Une prédominance dure des ombres : un fort contraste des ombres par rapport à l'arrière-plan dû à un fort éclairage des objets.
- Une prédominance douce des ombres : un faible contraste par rapport à l'arrière-plan dû à un faible éclairage vis-à-vis la position de l'objet.
- Un ombre (ou bien plusieurs) qui apparaît et disparaît suite au mouvement de l'objet exposé à la source lumineuse, ou bien suite à la variation de la source lumineuse elle-même.

L'objectif de cet hypothèse est de déterminer quelle méthode aura la meilleure capacité à différencier l'objet de son ombre, ou bien l'objet de l'ombre d'un autre objet, peu importe la prédominance de l'ombre (le minimum attendu de la méthode étant qu'elle fasse la distinction entre l'objet et l'ombre dans une prédominance dure). **L'hypothèse est que GMM aura une performance dégradée à séparer les objets des ombres par rapport YOLO. Autrement dit, GMM risque plus de confondre les objets et leurs ombres, voire même segmenter l'ombre en tant qu'un objet.**

L'intuition qui nous a amené à élaborer cette hypothèse est déjà mentionnée dans cet article [5]: "GMM algorithm cannot handle shadows".

2.3 Mouvement d'objets intermittents

Dans cette troisième hypothèse, on se place dans un contexte où on mesure la capacité d'une méthode à réagir avec le facteur occlusion. Dans cette perspective l'objet mis en question sera caché partiellement ou totalement par un autre objet dans une partie de la séquence des frames choisies. Dans cette optique, puisque GMM se base sur la soustraction d'arrière plan entre la frame courante et la moyenne des frames pour mesurer la variation des pixels d'un objet vis-à-vis de l'arrière-plan, tandis que YOLO se base sur une technique générant des probabilités de boîtes englobantes autour de l'objet susceptible d'être détecté puis sélectionnant la boîte la plus confiante, on présume que YOLO va rencontrer des difficultés à détecter l'objet intermittent vu qu'il apparaît puis disparaît partiellement ou en totalité selon le degré d'occlusion. Donc l'hypothèse est la suivante : **GMM doit avoir une meilleure performance que YOLO à détecter des objets intermittents sujets à l'occlusion.**

On résume les trois hypothèses dans le tableau suivant:

Table 1. Récapitulatif des trois hypothèses comparant GMM et YOLO

Cas d'utilisation	GMM	YOLO	Meilleure approche (théoriquement)
Mouvement d'objets avec arrière-plan dynamique	Mauvaise distinction entre l'arrière et l'avant plan. Confusion quasi-totale des objets avec les éléments en mouvement de l'arrière-plan (par exemple: les vagues dans une rivière).	Une meilleure distinction des objets détectés par rapport à leur environnement (arrière-plan en mouvement) grâce à la technique probabiliste de la boîte englobante la plus confiante.	YOLO
Mouvement d'objets avec des ombres à prédominance dure, douce et intermittente	Une forte possibilité que l'ombre soit détectée par GMM, surtout dans le cas d'une prédominance dure et/ou intermittente de cette ombre (calcul de la variation des pixels et donc une grande probabilité que l'ombre appartient à l'avant plan).	Peu importe la prédominance de l'ombre, YOLO est capable de bien discriminer la différence entre l'objet et l'ombre, s'il s'agit d'un objet flou en arrière-plan.	YOLO
Mouvement d'objets intermittent	Même si l'objet sera partiellement caché, GMM utilise les images précédentes de cet objet avec une moindre valeur de l'occlusion via la soustraction d'arrière-plan, et donc mémorise la variation des pixels pour une meilleure détection de cet objet sujet à une occlusion intermittente.	Une difficulté à détecter les objets séparément si le degré de superposition des objets est assez haut. Génère des boîtes englobantes autour de l'objet, donc si l'objet est caché, la boîte englobante risque d'encadrer à la même fois cet objet caché avec un autre objet.	GMM

III. DESCRIPTION DES EXPÉRIENCES, DE LA BASE DE DONNÉES UTILISÉE ET DES CRITÈRES D'ÉVALUATION

3.1 Description des expériences

Les expériences réalisées, pour vérifier les hypothèses précédentes, consistent à analyser 200 frames d'une vidéo (pour des raisons de temps d'exécution, nous avons choisi de limiter le nombre de frames plutôt qu'analyser la vidéo complète) correspondant à chaque hypothèse, et à calculer le score *Global-IoU* moyen de chaque méthode (décrit ci-dessous) pour cette analyse. Le score nous indiquera dans chaque cas quelle est la méthode la plus performante, et à quel point elle est plus efficace que la méthode concurrente.

En réalité, on ne va pas analyser toutes les frames d'une vidéo, mais seulement les frames pour lesquelles on dispose d'une vérité de terrain, permettant d'analyser les résultats de détection.

3.2 Présentation des données

Les données utilisées sont toutes issues du *dataset2014* de CDNET. Nous avons choisi une vidéo illustrant chaque cas que nous souhaitons étudier :

- ❑ Baseline : vidéo *PETS2006* : bien que cette vidéo ne soit associée à aucun cas, nous avons choisi de l'inclure pour illustrer les résultats "de base" des deux méthodes de détection (lien : <http://jacarini.dinf.usherbrooke.ca/static/dataset/baseline/PETS2006.zip>)
- ❑ Arrière-plan dynamique : vidéo *canoe* : une vidéo du passage d'un canot sur une rivière. Les remous et les vagues de la rivière constituent la partie dynamique de l'arrière-plan (lien : <http://jacarini.dinf.usherbrooke.ca/static/dataset/dynamicBackground/canoe.zip>)
- ❑ Ombres : vidéo *busStation* : une vidéo de personnes attendant à un arrêt de bus par temps ensoleillé. Leur ombre est visible sur le sol et les murs autour d'eux (lien : <http://jacarini.dinf.usherbrooke.ca/static/dataset/shadow/busStation.zip>)
- ❑ Occlusion : vidéo *sofa* : une vidéo d'un sofa et d'un homme retirant des objets posés dessus. L'homme réalise en passant devant les objets une occlusion de ceux-ci (lien : <http://jacarini.dinf.usherbrooke.ca/static/dataset/intermittentObjectMotion/sofa.zip>)

Pour le bon fonctionnement du code joint à ce rapport, télécharger et extraire les 4 datasets précédents dans le dossier "data" à la racine du projet.

3.3 Métrique d'évaluation

Une problématique causée par l'évaluation de ces deux méthodes résulte de la différence d'informations retournées par ces deux descripteurs. En effet, la soustraction d'arrière-plan renvoie un masque, une matrice indiquant pour chaque pixel de l'image s'il correspond à l'avant-plan ou à l'arrière-plan, tandis que Yolo renvoie une liste de boîtes englobantes correspondant aux objets détectés. Une première étape consiste donc à homogénéiser les résultats des deux méthodes pour mesurer leurs performances.

Pour cela, nous avons choisi de nous inspirer de la métrique IoU (*Intersection over Union*)¹, qui repose sur la comparaison de boîtes englobantes entre la détection et la vérité de terrain, en calculant la surface de l'intersection des boîtes sur la surface de l'union. Pour cela, nous avons utilisé une implémentation proposée sur StackOverflow permettant de tracer des boîtes englobantes sur un masque, et donc de récupérer ces boîtes pour appliquer IoU.

Cependant, cette métrique est parfois limitée, en particulier par exemple lorsqu'on détecte plus de deux boîtes englobantes sur une image (quelles boîtes associer entre la détection et la vérité de terrain) ou lorsque des objets se retrouvent dans la même boîte (deux personnes marchant côte à côte correspondant à deux boîtes pour Yolo, mais une seule région pour la détection d'arrière plan/la vérité de terrain). Pour répondre à ce problème, notre solution a été de ne considérer non pas une IoU sur les boîtes englobantes, mais directement sur la réunion des boîtes, tel qu'illustré sur les figures ci-dessous (Fig. 3.3.1): dans cette schématisation, les zones rouges correspondent aux boîtes englobantes de la vérité de terrain, et les bleues celles de la détection (Yolo ou Soustraction d'arrière-plan). On calcule alors la surface totale des intersections des deux zones de détection (en rose) et la surface totale de la réunion (en vert), et on peut alors obtenir une IoU "globale" pour la détection.

¹ <https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/>

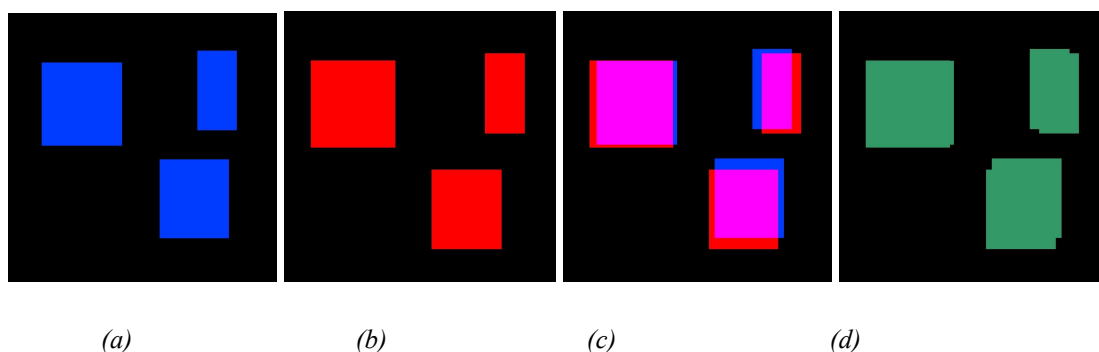


Figure 3.3.1. Description de la métrique Global-IoU: **a.** Boîtes englobantes de la vérité de terrain, **b.** Boîtes englobantes de la détection (YOLO ou GMM), **c.** intersection des boîtes englobantes de (a) et (b), **d.** réunion des boîtes englobantes de (a) et (b)

Cette métrique présente l'intérêt de ne pas dépendre du nombre total de boîtes englobantes, et du nombre de boîtes par objet, et de fonctionner comme IoU dans le cas d'une seule boîte pour chaque image. Par ailleurs, nous avons choisi de considérer, comme pour IoU classique, une "bonne" Global-IoU au delà d'une valeur de 0.5.

IV. DESCRIPTION DE L'IMPLÉMENTATION DES DEUX APPROCHES UTILISÉES

L'implémentation des approches a été réalisée en Python, et repose principalement sur les bibliothèques Numpy, Matplotlib, PIL et OpenCv.

4.1 Méthode de soustraction d'arrière-plan

Le code de la soustraction d'arrière-plan repose sur deux parties : dans un premier temps, la soustraction d'arrière-plan en elle-même, puis la génération de boîtes englobantes à partir de l'avant-plan détecté.

La soustraction d'arrière-plan implémente la méthode GMM présentée dans le cours², avec un facteur de sensibilité fixé empiriquement à 15. Par ailleurs, pour générer la moyenne utilisée comme arrière-plan, nous avons fait le choix d'utiliser les 100 premières frames d'une vidéo comme ensemble d'entraînement.

La génération des boîtes englobantes se base sur le post de Stackoverflow évoqué plus tôt³. En prenant en entrée une image en noir et blanc de l'avant-plan (avant-plan en blanc, arrière-plan en noir), on applique une érosion pour éliminer les pixels parasites, une dilatation pour démarquer les formes, on détecte alors grâce à OpenCv les contours autour desquels tracer les boîtes englobantes.

4.2 YOLO (You Only Look Once)

Pour cette méthode, nous avons repris l'implémentation de YOLOv3 qui a été proposée par [Adrian Rosebrock](#) (PyimageSearch) dans son tutorial [3], et qui été entraîné sur la base de données COCO de 80 « labels » (personnes, voitures, avions...).

D'une manière générale, YOLO utilise la stratégie de la détection à un étage (one-stage detector strategy), par le biais d'une grille à un nombre fixe de prédictions. Ainsi, notre problème de détection d'objet sera traité en tant que problème de régression : à partir d'une image « entrée », on détermine simultanément les boîtes englobantes et les probabilités d'étiquettes de classe correspondantes.

² <https://colab.research.google.com/github/gabilodeau/INF6804/blob/master/SingleGaussianBGS.ipynb>

³ <https://stackoverflow.com/questions/60646384/python-opencv-background-subtraction-and-bounding-box>

Notre implémentation de Yolo repose sur deux paramètres (les plus importants) :

- **confidence** = 0.5 : représente la probabilité minimale pour filtrer les faibles détections, en d'autres termes les détections qui ont une probabilité inférieure à 0.5 seront ignorées. Le choix de la valeur 0.5 n'est pas arbitraire (cette valeur a été standardisée dans la recommandation de « PASCAL VOC Challenge »).
- **threshold** = 0.3 : ce seuil est fixé pour la méthode « Non Max Supression ». Lorsqu'un objet est susceptible d'être potentiellement détecté, plusieurs boîtes englobantes sont générées pour encadrer cet objet. Cette méthode, via la valeur de « threshold », permet de choisir une seule boîte englobante (celle qui a le plus de probabilité de correspondance avec l'objet détecté).

Après avoir créé une instance du modèle YOLO pré-entraînée sur un réseau de neurones profond (DNN) avec les poids et les paramètres de configuration déjà donnés par ce modèle, on extrait la liste des boîtes englobantes (d'une manière plus précise, on va les appeler boîtes englobantes potentielles) qui seront stockées dans une liste « boxes ». Cette extraction se fait à un premier niveau en appliquant une opération de mise-à-échelle (scaling) entre les coordonnées des boîtes englobantes et la taille de l'image elle-même, puis à un deuxième niveau en utilisant le centre de cette boîte englobante afin de déterminer les coordonnées du coin supérieur gauche de cette dernière.

Puisque nous obtenons des boîtes englobantes potentiellement superposées, on applique la technique de « Non Max Suppression » pour ne garder que les boîtes englobantes les plus probables selon la valeur fixée de « confidence » et « threshold ».

S'il existe au moins une boîte englobante fiable, on la dessine avec une couleur de cadre choisie arbitrairement, en lui associant la classe COCO qui lui est appropriée (label) et le score de confiance adéquat, pour afficher ultimement notre image avec chaque objet détecté encadré par la boîte englobante la plus probable, la classe de cet objet et le score de confiance.

Pour le bon fonctionnement du code joint à ce rapport, télécharger et extraire à la racine du projet les données nécessaires à Yolo à l'adresse suivante :

<https://s3-us-west-2.amazonaws.com/static.pyimagesearch.com/opencv-yolo/yolo-object-detection.zip>.

4.3 Métrique Global-IoU

La métrique Global-IoU ayant été imaginée pour ce TP, son implémentation est entièrement originale. Considérant une liste de boîtes englobantes et la taille de l'image d'entrée, on va générer une matrice de booléens de la taille de notre image, chaque case décrivant un pixel. Une case de cette matrice aura pour valeur *True* si le pixel associé est recouvert par au moins une des boîtes englobantes, et *False* dans le cas contraire.

On a alors pour chaque entrée, Détection ou Vérité de Terrain, une matrice indiquant la zone globale détectée. Dès lors, il est très simple de calculer l'intersection et l'union : un *Et* logique pour l'intersection, un *Ou* logique pour l'union. Il suffit alors de compter le nombre de *True* dans ces deux matrices pour avoir la surface des zones associées, et donc la Global-IoU.

V. PRÉSENTATION DES RÉSULTATS DE TEST

5.1 Résultats de détection

Chaque figure correspond à la détection sur l'une des frames de la vidéo. Les boîtes englobantes sont tracées autour des régions identifiées. Le *Groundtruth* fourni avec les données contient les zones d'intérêt à identifier (encadrées en rouge), et à comparer avec les régions détectées par GMM (en bleu) et par Yolo (couleurs variables).

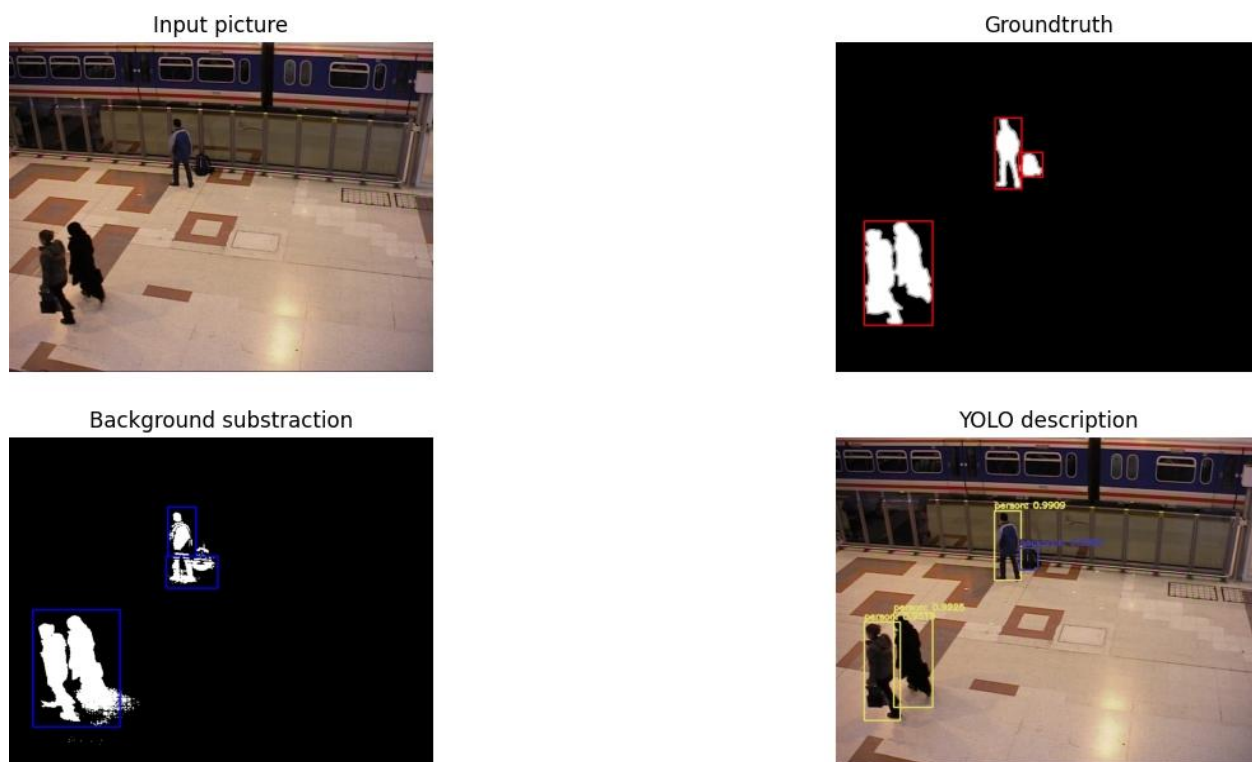


Figure 5.1.1 : Détection sur PETS2006



Figure 5.1.2 : Détection sur Canoe



Figure 5.1.3 : Détection sur BusStation



Figure 5.1.4 : Détection sur sofa

5.2 Mesure des Global-IoU

On rassemble les mesures de Global-IoU pour chaque vidéo dans le tableau suivant :

Table 2. Mesures du Global-IoU pour GMM et YOLO des différents cas d'utilisation

Vidéo	Global-IoU (GMM)	Global-IoU (Yolo)
PETS2006	0.56	0.77
Canoe	0.0	0.39
BusStation	0.53	0.76
Sofa	0.49	0.28

VI. PRÉSENTATION DES RÉSULTATS DE TEST

6.1 Mouvement d'objets avec arrière-plan dynamique

Comme supposé dans la première hypothèse, Yolo est effectivement bien plus performant que GMM pour la soustraction d'arrière-plan, avec une IoU moyenne de 0.39. GMM de son côté ne parvient pas à détecter une quelconque zone d'intérêt ; cela est certainement dû, comme supposé, au fait que la méthode ne parvient pas à générer d'arrière-plan moyen si les arrière-plans présentent trop de variations, comme ici les remous de l'eau, et donc à détecter des variations sur cet arrière-plan.

6.2 Mouvement d'objets avec des ombres à prédominance dure, douce et intermittente

Les expériences ont montré que dans une scène extérieure en plein air (*outdoor-environment*), GMM trouve plus de difficultés que YOLO à séparer les objets (des passagers qui attendent près d'une station de bus) de leurs ombres, surtout avec des images plus sombres où la prédominance des ombres est dure et intermittente, générant ainsi une précision relativement correcte pour isoler les ombres de leurs objets (Global-IoU de 0.53). Cependant, YOLO a réussi à montrer une performance bien meilleure dans ce cas (Global-IoU de 0.76) peu importe les changements dans la scène au niveau de l'exposition de l'objet à la source lumineuse, générant ainsi des ombres à prédominance dure, douce et intermittente. Cela est tout à fait cohérent avec notre intuition qui a été mise en valeur dans l'hypothèse 2, car les variations de contraste pour les pixels (même au niveau des ombres) créent de la confusion pour GMM qui détecte ainsi les ombres comme des "faux objets", alors que YOLO répond assez bien à ce défi, le score de confiance de la boîte englobante utilisée dans YOLO étant assez faible pour détecter cette ombre en tant qu'un objet.

6.3 Mouvement d'objets intermittents




Comme supposé dans la troisième hypothèse, Yolo rencontre des difficultés à détecter les objets intermittents par rapport à GMM. D'une manière plus précise, contrairement à GMM qui performe relativement bien vu la complexité du défi (Global-IoU de 0.49), YOLO trouve des difficultés à maintenir un score de confiance assez élevé pour la boîte englobante qui encadre l'objet à détecter, vu l'intermittence qui engendre l'occlusion vis-à-vis cet objet en question. Dans notre cas, l'objet en question est le sac marron à l'extrémité droite sur le sofa (Figure 5.1.4) qui apparaît initialement en totalité, puis caché par la personne en mouvement tantôt partiellement, tantôt totalement, jusqu'à réapparaître en totalité. Le résultat obtenu est expliqué par le fait que la

boîte englobante de YOLO confond l'objet en état d'intermittence (le sac marron) sur un volet de confusion qui fonctionne parallèlement d'une part avec le sofa, et d'autre part par une partie de l'individu en mouvement qui crée de l'occlusion par rapport à ce sac (que se soit totale ou bien partielle).

Ainsi, YOLO donne un Global-IoU qui est égal à peu près la moitié de celui de GMM (Global-IoU de 0.28). Le résultat obtenu est concordant avec notre intuition de l'hypothèse 3, et cela est tout à fait axiomatique vu que GMM se base sur la détection de la variation des contrastes des pixels de l'objet entre le moment actuel et une moyenne des frames, même si l'objet en question est en état d'intermittence.

A titre de retour sur hypothèses, nous élaborons ce tableau comme un récapitulatif.

Table 3. Retour sur hypothèses et récapitulatif des résultats obtenus (théoriques et pratiques)

Cas d'utilisation	Meilleure approche (théoriquement)	Global-IoU	Correspondance entre la théorie et la pratique
Mouvement d'objets avec arrière-plan dynamique	YOLO	0.39	
Mouvement d'objets avec des ombres à prédominance dure, douce et intermittente	YOLO	0.76	
Mouvement d'objets intermittent	GMM	0.49	

CONCLUSION

Comparer l'approche de soustraction d'arrière-plan avec celle de détection par classification représente un vrai défi de recherche jusqu'à l'instant.

Via ce TP, nous avons choisi de comparer une variante de chaque méthode qui sont GMM et YOLO via une analyse qui nous a mené à définir Global-IoU comme nouvelle métrique de comparaison de performance de ces deux variantes. Les résultats obtenus nous ont permis de tirer ces conclusions:

D'une part, on ne peut pas conclure dans l'absolu que YOLO est plus performant que GMM. En effet, avec notre choix des trois cas d'utilisation, YOLO a donné une meilleure performance avec le cas de mouvement d'objet en arrière-plan dynamique et celui dont l'objet en mouvement avec des ombres à prédominance dure, douce et intermittente. D'autre part, on ne peut pas aussi affirmer que l'approche de soustraction d'arrière-plan présente de meilleurs résultats que celle de détection par classification (peu importe les variantes utilisées dans ces deux approches).

Dans notre cas, nous avons choisi de faire la comparaison entre YOLO et GMM, mais d'autres variantes pourraient donner un résultat différent du celui qu'on a obtenu selon les paramètres des méthodes ou le choix des cas d'utilisations à traiter. Cependant, une autre approche pourrait être particulièrement pertinente : au lieu de comparer les deux méthodes, on pourrait chercher à les combiner, pour tirer parti des forces de chacune de ces techniques.

BIBLIOGRAPHIE ET RÉFÉRENCES

- [1] Redmon, Joseph, et al. "You Only Look Once: Unified, Real-Time Object Detection." *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 9 May 2016, doi:10.1109/cvpr.2016.91.
- [2] Redmon, Joseph, et al. "YOLOV3: An Incremental Improvement." *2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 9 Apr 2018, doi:1804.02767
- [3] Adrian Rosebrock. "YOLO Object Detection with OpenCV." *PyImageSearch*, 18 Apr. 2020, www.pyimagesearch.com/2018/11/12/yolo-object-detection-with-opencv/.
- [4] Peng, Qiwei, et al. "Pedestrian Detection for Transformer Substation Based on Gaussian Mixture Model and YOLO." *2016 8th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, 15 Dec. 2016, doi:10.1109/ihmsc.2016.130.
- [5] Kim, Chulyeon, et al. "A Hybrid Framework Combining Background Subtraction and Deep Neural Networks for Rapid Person Detection." *Journal of Big Data*, vol. 5, no. 1, 10 July 2018, doi:10.1186/s40537-018-0131-x.