REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE MOHAMED BOUDIAF - M'SILA

FACULTE DE TECHNOLOGIE

DEPARTEMENT D'ELECTRONIQUE

N° :….

DOMAINE : SCIENCES ET TECHNOLOGIES

FILIERE : TELECOMMUNICATIONS

OPTION : Systèmes des Télécommunications

Mémoire présenté pour l'obtention Du diplôme de Master Académique

Intitulé

Détection d'interaction humaine-objet à l'aide du deep learning

Préparés par :

Benslimane Manel

Halitim Amira

Les Membres du Jury :

Président	**Dr ZERDOUMI Zohra**	**U. BOUDIAF Mohamed. M'sila.**
Encadreur	**prof LALAOUI Lahouaoui**	**U. BOUDIAF Mohamed. M'sila.**
Examinateur	**Dr KHALFA Ali**	**U. BOUDIAF Mohamed. M'sila.**

Année universitaire : 2024 / 2025

REMERCIEMENTS

En commençant ce travail, il nous tient à cœur d'adresser nos sincères remerciements à toutes les personnes qui ont contribué, de près ou de loin, à la réalisation de ce projet.

Nous tenons tout d'abord à remercier notre encadrant, M. LALAOUI Lahouaoui, pour sa disponibilité, ses conseils avisés et son accompagnement tout au long de cette expérience. Son soutien a été précieux, aussi bien dans les moments de doute que dans les instants de réussite.

Nos remerciements vont également à notre sous-encadrant, M. DJALAB Abdelhak, pour sa précieuse aide, ses orientations pertinentes et sa bienveillance.

Nous exprimons aussi notre profonde gratitude à Mme ZERDOUMI Zohra, présidente du jury, ainsi qu'à M. KHALFA Ali, examinateur, pour le temps qu'ils ont consacré à l'évaluation de notre travail et pour leurs remarques enrichissantes.

Nous remercions chaleureusement l'ensemble de l'équipe pédagogique du département d'électronique pour la qualité de l'enseignement dispensé et pour nous avoir permis de grandir, tant sur le plan académique que personnel.

Enfin, nous n'oublions pas nos familles et nos amis, pour leur soutien indéfectible, leur patience et leurs encouragements constants tout au long de notre parcours.

À toutes et à tous, merci du fond du cœur.

Dédicace

Je dédie ce travail à :

A l'âme de mon cher père Ben Slimane smaine, J'espère que tu habites les paradis.

A ma mère, qui m'a encouragé à aller de l'avent et qui m'a donnée tout son amour pour reprendre mes études.

A ma sœur et mes frères.

A toute ma famille sans aucune exception ;

A tous mes professeurs et enseignants durant tout mon cursus universitaire et qui m'ont permis de réussir dans mes études.

A tous mes amis.

A toute personne ayant contribué à ce travail de près ou de loin.

Ben slimane Manel

Dédicace

À ceux qui, après Dieu, m'ont aidé à atteindre ce moment

À ma mère, Halitim moubaraka, qui m'a appris que la volonté fait des miracles.

À mon père, Halitim Hamoudi, mon modèle,

À mon premier soutien, mon deuxième père, Halitim Menad,

Ainsi qu'à mes frères, sœurs et fiancé,

Mes amis et tous ceux qui me font sourire.

À vous tous, je dédie le fruit de mes années d'efforts et de labeur.

Halitim Amira

Résumé

La détection d'interaction humain-objet, un domaine clé de la vision par ordinateur, en mettant l'accent sur les avancées apportées par les techniques de Deep learning. Il explore diverses architectures de réseaux de neurones, telles que R-CNN et YOLO, et analyse leur efficacité dans l'identification et le suivi d'objets dans des environnements complexes. En abordant les défis liés à la détection, comme les occlusions et les variations d'éclairage, le mémoire démontre l'importance d'une approche intégrée pour améliorer la précision et la fiabilité des systèmes de détection dans des applications pratiques.

Abstract

Human-object interaction detection, a key area of computer vision, with a focus on the advances brought by deep learning techniques. It explores various neural network architectures, such as R-CNN and YOLO, and analyzes their effectiveness in identifying and tracking objects in complex environments. By addressing detection challenges, such as occlusions and lighting variations, the thesis demonstrates the importance of an integrated approach to improve the accuracy and reliability of detection systems in practical applications.

ملخص

كشف التفاعل بين الإنسان والأشياء، وهو مجال رئيسي في مجال الرؤية الحاسوبية، مع التركيز على التطورات التي جلبتها تقنيات التعلم العميق. ويستكشف هذا الكتاب مختلف هياكل الشبكات العصبية، مثل R-CNN وYOLO، ويحلل مدى فعاليتها في تحديد وتتبع الكائنات في البيئات المعقدة. ومن خلال معالجة تحديات الكشف مثل الانسدادات واختلافات الإضاءة، يوضح البحث أهمية اتباع نهج متكامل لتحسين دقة وموثوقية أنظمة الكشف في التطبيقات العملية.

Mots-clés : Détection d’interactions humain-objet (HOI), Apprentissage profond, Architecture de réseaux de neurones, YOLOv5, Faster R-CNN, SSD, U-Net++.

Table des matières

Liste des abréviations :

R-CNN : Réseau de neurones convolutif basé région.

SVM : Support Vector Machine (machine à vecteurs de support).

RPN : Réseau de Proposition de Région.

FCN: Fully Convolutional Network.

SSD: Single Shot MultiBox Detector.

YOLO: You Only Look Once.

LKC-Net: Large Kernel Convolution Object Detection Network.

HRNet: haute résolution network.

HOI: human-object interaction.

NLP: traitement du Langage naturel.

QPIC: Query-Based Pairwise Interaction and Context-aware model**.**

DETR-like: detection transformer like.

Liste des figures :

Liste des tableaux :

Introduction générale

Introduction générale

La capacité des machines à percevoir et comprendre leur environnement est au cœur des avancées récentes dans le domaine de l'intelligence artificielle. Parmi les nombreux champs d'étude qui s'y rattachent, la détection d'interactions humain-objet (Human-Object Interaction, HOI) suscite un intérêt croissant. Elle joue un rôle central dans le développement de systèmes intelligents capables d'interagir de manière naturelle et contextuelle avec leur environnement, que ce soit dans des applications de robotique autonome, de vidéosurveillance intelligente, de réalité augmentée, ou encore d'interfaces homme-machine.

Grâce aux progrès fulgurants du deep learning, les méthodes de détection d'objets ont connu des améliorations significatives, tant en termes de précision que de vitesse d'exécution. Ces avancées permettent non seulement d'identifier des objets dans des images ou des vidéos, mais aussi de reconnaître les relations dynamiques entre un individu et les objets avec lesquels il interagit. Identifier qu'une personne "tient", "pousse", ou "utilise" un objet devient ainsi une compétence essentielle pour de nombreuses applications modernes.

Problématique

Cependant, malgré ces avancées, de nombreux défis subsistent. Les interactions humain-objet sont complexes, contextuelles et souvent ambiguës, ce qui rend leur détection difficile dans des environnements réels et non contrôlés. De plus, la variabilité des postures humaines, les occlusions, la diversité des objets et les contraintes de calcul en temps réel posent des problèmes importants pour les systèmes actuels.

Comment concevoir et mettre en œuvre un système efficace de détection d'interactions humain-objet, capable de fonctionner de manière robuste dans des conditions réelles, tout en exploitant les architectures de deep learning les plus performantes ?

Ce mémoire se propose de répondre à cette question en explorant les fondements théoriques et techniques de la détection HOI. Il s'appuie sur une revue approfondie des approches existantes, une analyse comparative des architectures de détection d'objets, et une expérimentation concrète à l'aide de réseaux de neurones convolutifs avancés. L'objectif est de mettre en lumière les meilleures pratiques actuelles et de proposer des pistes d'amélioration pour les systèmes de détection à venir.

Ce mémoire se propose d'explorer les fondements théoriques et pratiques de cette discipline, en analysant les défis et les opportunités qu'elle présente.

Chapitre I : Ce chapitre introduit les concepts fondamentaux de la détection d'interaction humain-objet, en définissant les termes clés et en présentant les différentes approches utilisées dans ce domaine. Nous y abordons les techniques de modélisation d'objets, les caractéristiques essentielles à prendre en compte, et les méthodes de détection basées sur des modèles de mouvement. Ce chapitre pose les bases nécessaires pour comprendre les enjeux et les applications de la détection d'objets dans des contextes variés.

Chapitre II : Dans ce chapitre, nous examinons les différentes méthodes de classification utilisées pour la détection d'objets, en mettant l'accent sur les réseaux de neurones convolutifs (CNN) et leurs variantes. Nous analysons des architectures spécifiques telles que R-CNN, Fast R-CNN et YOLO, en discutant de leurs avantages et inconvénients. Ce chapitre illustre comment ces techniques permettent d'améliorer la précision et la rapidité de la détection, tout en abordant les défis associés à leur mise en œuvre.

Chapitre III : Le dernier chapitre se concentre sur la conception et l'implémentation des systèmes de détection d'objets, en détaillant l'environnement de travail et les bibliothèques utilisées. Nous y présentons les résultats obtenus lors des expérimentations, en comparant les performances des différentes architectures choisies, telles que YOLOv5 et Faster R-CNN. Ce chapitre met en évidence l'importance d'une approche méthodique pour optimiser les systèmes de détection et souligne les perspectives d'amélioration future.

Chapitre I : La détection d'interaction humain-objet

I.1 Introduction

Les usages de la détection d'objets sont variés. On peut notamment mentionner le suivi d'élément s, l'analyse des comportements, la compression de vidéo, etc. Les méthodes de détection du mou vement peuvent être perçues comme une étape préalable visant à diminuer le volume d'informati ons à examiner. Un objet en déplacement est identifié si sa position évolue par rapport à celle d'u n groupe d'objets immobiles ou s'il est repéré dans une image capturée à un instant t de la séquen ce, à un endroit distinct de celui qu'il occupait dans l'image précédente. Néanmoins, le suivi repr ésente un défi plus ardu que la détection, car il faut tenir compte d'autres contraintes significative s, notamment le fait que certains objets peuvent disparaître temporairement du champ de vision d e la caméra avant de réapparaître lorsqu'ils sont obstrués par un élément ou une personne.

Dans ce chapitre nous avons présenté d'une façon générale la détection d'objet, leur définition, pourquoi nous utiliserons, les différentes méthodes utilisées et leurs domaines d'applications.

I.2 définition de deep Learning

Le Deep Learning, ou apprentissage avancé, peut être perçu comme appartenant à une catégorie plus vaste de Machine Learning, qui permet aux dispositifs informatiques de convertir des notion s plus élémentaires en idées plus abstraites et élaborée. [2]

I.3 Définition de détection d'objet

La détection d'objets est une tâche de vision par ordinateur qui vise à localiser les objets présents dans les images numériques. Il s'agit d'un exemple d'intelligence artificielle qui consiste à entraîner les ordinateurs à voir comme les humains, notamment en reconnaissant et en classant les objets par catégorie sémantique.[1] La localisation d'objet est une technique qui permet de déterminer la position d'un objet dans une image en le délimitant à l'aide d'une boîte englobant. La classification des objets est une autre technique qui permet de déterminer la catégorie à laquelle appartient l'objet détecté. La tâche de détection d'objets associe des sous-tâches de localisation et de classification d'objets pour estimer simultanément l'emplacement et le type des instances d'objets dans une ou plusieurs images. [3]

I.4 Modélisation d'un objet

I.4.1 Caractéristiques d'un objet

Dans le domaine du mouvement d'un objet dans une séquence, un objet peut avoir différentes caractéristique, il peut être :

– **Constant ou variable :** La forme, le mouvement, les couleurs ou les textures d'un objet varient ou non au cours du temps. Ces considérations jouent en faveur de l'adoption ou non d'un modèle de l'objet pour en améliorer le suivi.

– **Rigide ou non rigide :** pour la forme nous parlerons d'objet rigide (voiture) ou non-rigide (corps humain) c.à.d.la distance entre les points de l'objet est constante ou variable au cours du temps.

– **Unique ou multiple :** Dans le cadre d'un suivi labellisé où chaque objet porte une étiquette et n'en change pas durant l'opération de son suivi, une distinction entre chaque objet est nécessaire c.à.d. que chaque objet doit être unique. L'unicité repose, là aussi, sur un modèle. Cependant celui-ci n'est pas nécessairement connu au préalable. Cependant une connaissance des critères potentiellement discriminants permet une amélioration de la charge de calcul. La description de chaque objet selon des critères pertinents assure une labellisation fiable. [4]

I.4.2 Représentation d'un objet

Un objet, dans un scénario de suivi, est une entité indépendante pourvue de son identité spatiale (sa forme, son contour,...etc.) dans un environnement particulier. Cependant, on ne dispose pas toujours de toutes les informations qui caractérisent l'objet à suivre, mais on utilise un modèle de représentation de son état à surveiller. Une modélisation d'objets par catégories peut être présentée. [4]

I.4.2.1 Modélisation par points

L'objet à suivre est représenté soit par un point qui représente le centre de gravité de l'objet concerné, soit par un ensemble de points [4], (Figure.1). En général cette dernière est efficace pour le suivi d'objets de petites tailles ou régions.

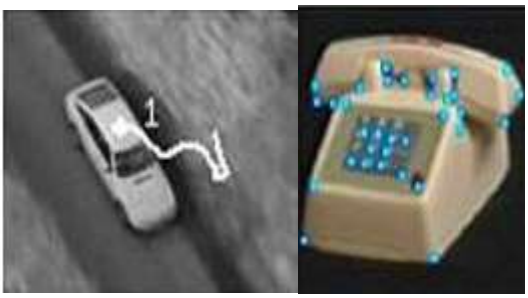

Figure I.1: Points de contrôle.

I.4.2.2 Modélisation par formes géométriques

La forme d'un objet est représentée par un rectangle ou (boîtes englobantes) (Figure.2) ou une ellipse, le mouvement d'objet défini par cette représentation est souvent modélisé par une translation, projection et autre transformation, cependant ces formes géométriques sont plus appropriées à représenter les objets rigides simples, elles sont également employées pour suivre les objets souples. [4]

Figure I.2: Boîtes englobantes.

I.4.2.3 Modélisation par silhouette et contour :

Par définition, un contour est la frontière qui sépare deux objets dans une image. La région à l'intérieur du contour s'appelle la silhouette de l'objet (Figure.3). La représentation par le contour et la silhouette est appropriée pour représenter des objets non rigides et les modèles déformables complexes. [4]

Figure I.3: Boîtes englobantes.

I.4.2.4 Modélisation par des modèles de formes articulées :

Les objets articulés sont composés de parties du corps qui sont liés avec des joints. Par exemple le corps humain est un objet articulé avec torse, jambes, mains, tête et pieds reliés par des joints (Figure.4). On peut modéliser ces parties en utilisant des cylindres ou des ellipses, les rapports entre eux sont régis par des modèles de cinématique de mouvement, par exemple l'angle commun etc. [4]

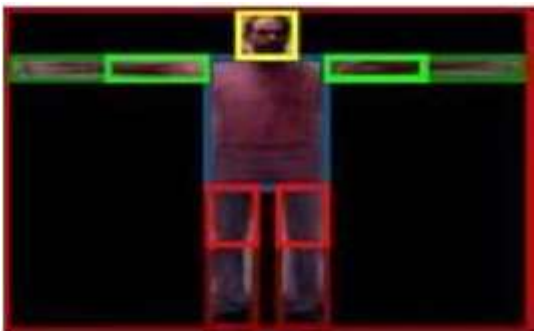

Figure I.4: Modèle d'apparence articulé.

I.4.2.5 Modèles squelettiques

Le squelette d'objet peut être extrait en appliquant la transformation de l'axe médiane à la silhouette d'objet (Figure.5).Ce modèle est généralement utilisé pour la reconnaissance de l'objet et du suivi des modèles cinématiques complexes et la reconnaissance de mouvement. [4]

Figure I.5: Modèle d'apparence articulé.

I.5 Pour quoi la détection d'objets

La détection d'objets est un secteur extrêmement vaste et crucial dans le domaine de la recherche, car les travaux contemporains visent à développer des systèmes qui s'apparentent aux aptitudes humaines en matière de perception, de suivi et d'identification des objets. L'importance de la reconnaissance provient du fait qu'un bon résultat à cette étape permet d'obtenir une reconnaissance efficace, et cette phase est également complexe à gérer en raison de divers défis tels que la dimension des objets détectés, les conditions d'éclairage, la forme, la grande diversité des objets, la rapidité de réponse en temps réel, ainsi que la complexité des arrière-plans… etc. La reconnaissance d'objets est l'une des applications pratiques les plus captivantes dans la vie quotidienne ; par exemple, elle est employée dans la surveillance du trafic ou pour des véhicules dotés d'une assistance à la conduite automatique ou partielle. Cette dernière est utilisée par des entreprises pour évaluer la qualité des produits fabriqués, en déterminant lesquels sont conformes et lesquels ne le sont pas (par exemple, une société qui produit des pièces et dont le système va contrôler la qualité).

I.6 Les méthodes de détection du mouvement

I.6.1 La différence temporelle des images

La différence temporelle détecte la région de mouvement grâce à la différence de pixel par pixel de deux trames consécutives dans un flux vidéo [5] [6]. Cette méthode adapte le changement de la scène. Mais elle est moins d'efficacité parce que dans une durée du temps Δt, peut-être, on détecte seulement une partie d'objet, par exemple : la main, la tête etc. Et le vide (la région où l'objet s'est déplacé l'autre lieu) est aussi détecté. Dans ce cas-là, c'est très difficile à extraire des propriétés de mouvement (la taille, la position, la vitesse etc.) et à suivre le mouvement [7].

(a) (b)

Figure I.6: Un exemple de la différence temporelle. (a) une scène simple avec deux objets, (b) les régions rouges sont la différence entre deux images consécutives.

Voici, c'est l'idée principale de cette méthode : Soit I_t l'image à l'instant t et I_{t-1} l'image à l'instant t-1. L'objet du mouvement se compose les pixels qui satisfont l'équation suivante :

$$max(|I_t(x,y) * c - I_{t-1}(x,y) * c|, c = (R,G,B)) >= Seuil \quad (1)$$

Comme ci-dessus a déjà expliqué, on peut améliorer cette méthode en façon suivante au lieu de soustraire l'image à l'instant t-1, on soustrait la moyenne de N images dernières [6]. Soit I_{tm} la moyenne de N dernières images à l'instant t. L'objet du mouvement se compose les pixels qui satisfont l'équation suivante :

$$max = (|I_t(x,y) * c - I_{tm}(x,y) * c|, c = (R,G,B)) >= Seuil \quad (2)$$

La moyenne de N images à l'instant t +1 est mise à jours :

$$I_{t+1}(x,y) * c = \alpha I_t(x,y) * c + (1-\alpha) I_{tm}(x,y) * c, c = R,G.B) \quad (3)$$

Où α ∈(0,1) est une constante et est décidé par la pratique.

Les avantages :

- Adapte le changement de la scène.
- Détecte la région de mouvement.

Les inconvénients :

- Ne permet pas de détecter le mouvement dans les zones uniformes intérieures à l'objet
- Ne fonctionne pas dans plusieurs cas, pour différentes raisons, telles que : la présence de bruit du capteur et les changements de luminosité de la scène qui modifient les intensités des pixels.

I.6.2 Double de différence temporelle et caractère de contour :

Dans cette manière, on utilise aussi la différence de pixel par pixel des trames consécutives dans un flux vidéo comme la 1ère méthode. Mais on va utiliser trois trames consécutives [8].

Cette façon nous donne le résultat meilleur que celui de la 1ère méthode, tandis qu'elle adapte aussi le changement de la scène. Voici, c'est l'idée principale de cette méthode : Soit I l'image à l'instant t, I l'image à l'instant t-1 et I l'image à l'instant t-2.

L'objet du mouvement se compose les pixels qui satisfont l'équation suivante :

$$I_1(x,y) = max\big(|I_t(x,y) * c - I_{t-1}(x,y) * c|, c = (R,G,B)\big) >= seuil \quad (4)$$

$$I_2(x,y) = max\big(|I_{t-1}(x,y) * c - I_{t-2}(x,y) * c|, c = (R,G,B)\big) >= seuil \quad (5)$$

$$I_{resultat(x,y)} = I(x,y)I(x,y) \quad (6)$$

Nous extrayons la région des objets mouvants par la double méthode de différence. Dans le cas, l'objet se déplace lentement ou il n'y a qu'une partie d'objet se déplace, nous ne pouvons pas obtenir complètement la forme d'objet. Peut-être, un objet sera divisé en plusieurs régions.

Nous utilisons donc le caractère de contour pour combiner ces régions. Après avoir masqué l'image à l'instant t-1 avec la région mouvante obtenue à partir de la double méthode de différence, le contour est calculée dans la région où le mouvement est produit.

Le contour est calculé à partir de l'image F qui est l'image à l'instant t-1 masquée avec R = I1 (x, y) U I2 (x, y). Le contour est représenté par :

$$G = \begin{bmatrix} F_x^2 & F_xF_y \\ F_xF_y & F_x^2 \end{bmatrix} \quad (7)$$

En utilisant un caractère de gradient avec l'intensité de l'image masquée F. En chaque pixel que le mouvement est détecté par l'intermédiaire de la double méthode de différence dans l'armature courante, le coefficient est calculé. C'est-à-dire, les pixels commandés par le caractère maximal de contour sont choisis en tant que points de caractère de contour parce que les valeurs propres minimum plus grandes sont les caractères plus forts de contour.

Avec cette façon, à l'instant t, on va obtenir des objets mouvant à l'instant t-1. C'est à-dire, on est en retard de 0,5 seconde. Autre chose, si l'objet ne se déplace pas pendant quelques secondes, cette manière ne peut pas détecter le mouvement. Dans ce cas, nous utilisons le contour d'objet avec le résultat de dernière étape pour détecter le mouvement. [6]

Les avantages :

- Le résultat de cette méthode est meilleur par rapport la méthode précédant.
- Détecte la région de mouvement.
- Utilise trois trames.

Les inconvénients :

- Si l'objet ne se déplace pas pendant quelques secondes, cette manière ne peut pas détecter le mouvement.

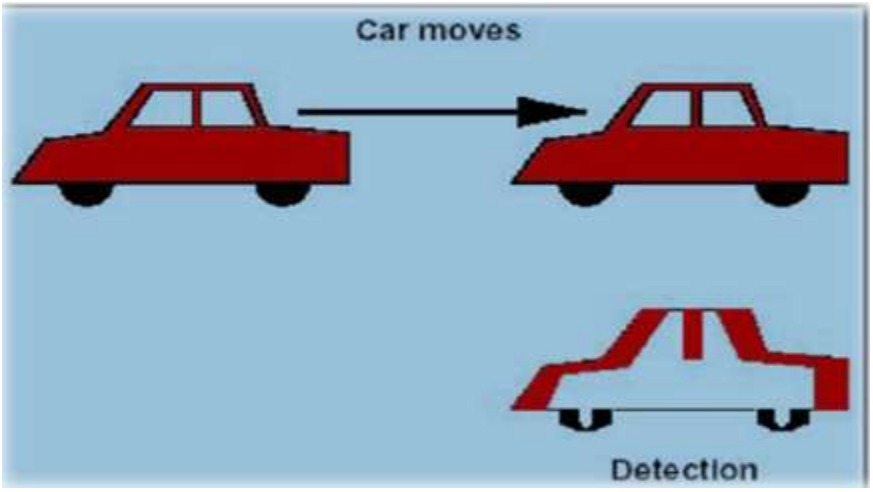

Figure I.7: Détection de mouvement par différence temporel.

I.6.3 La soustraction de l'image de fond :

La troisième méthode, on utilise une image de fond. Cette méthode est très populaire et elle est utilisée par plusieurs applications. Cette façon détecte la région de mouvement en soustrayant pixel par pixel l'image courante à l'image de fond.

Figure I.8: Exemple soustraction du fond.

I.6.3.1. Algorithme :

Soit I t l'image à l'instant t. B t est l'image de fond à l'instant t. L'objet du mouvement se compose les pixels qui satisfont l'équation suivante :

$$max = |I_t(x,y) * c - B_t(x,y)|, c = (R,G,B)) >= S(x,y) \quad (8)$$

Où St (x, y) est seuil de la position (x, y) à l'instant t. Cette méthode nous permet obtenir la forme complète d'objet et compter exactement les objets du mouvement parce que l'image de fond n'a pas l'objet mouvant. Cependant, en réel, le changement de l'espace a lieu souvent (Spécialement, c'est le changement de la lumière : le matin, le soir, il y a du soleil, il pleut etc.) [7].

Cela influence le résultat si l'image de fond n'est pas mise à jour. Ou, dans un autre cas, l'objet stoppe dans une durée du temps long.

On a besoin de mettre à jours cet objet à l'image de fond. Evidemment, on a besoin d'un seuil pour déterminer qu'un objet va additionner de l'image de fond s'il a stoppé dans N trames. Alors, on doit mettre à jours l'image de fond.

Selon Yiğithan Dedeoğlu [9], il y a une façon simple pour mettre à jours l'image de fond et la matrice de seuil : soit O l'ensemble de l'objet mouvant.

$$B_{t+1}(x,y) = \begin{cases} \alpha B_t(x,y) + (1-\alpha) I_t(x,y), & (x,y) \epsilon B \\ \beta B_t(x,y) + (1-\beta) I_t(x,y), & (x,y) \epsilon O \end{cases} \quad (9)$$

$$S_{t+1}(x,y) = \begin{cases} \alpha S_t(x+y) + (1-\alpha)(\gamma x |I_t(x,y) - B_t(x,y)|), & (x,y) \epsilon B \\ S_t(x,y), & (x,y) \epsilon O \end{cases} \quad (10)$$

Où α, β ∈(0,1) sont les constantes et sont décidés par la pratique. γ est le taux d'influence de la différence entre l'image et le fond. Il est aussi décidé par la pratique.

Un problème, comment on peut choisir α, β et γ pour tout le cas ? C'est très difficile. Ainsi, dans la partie pratique de ce TIPE, j'ai utilisé une méthode qui est plus simple que la méthode de Yiğithan Dedeoğlu [8]. Je mets à jours seulement l'image de fond en manière suivante :

$$B_{t+1}(x,y) = | \begin{cases} B_t(x,y), & (x,y) \epsilon O \\ I_t(x,y), & (x,y) \epsilon B \end{cases} \quad (11)$$

Les Avantages :

- Cette méthode est très populaire.

- Utilisée par plusieurs applications.
- Détecte l'objet complèteme.

I.7 Domaine d'utilisation :

La détection des interactions entre humains et objets (Human-Object Interaction, HOI) est un secteur de la vision par ordinateur qui cherche à saisir comment les individus interagissent avec les objets présents dans une image ou une vidéo. Les applications sont multiples et touchent divers domaines :

I.7.1 Surveillance et sécurité :

- Identification des comportements suspects (vol, agression, intrusion).
- Surveillance intelligente dans des espaces publics, aéroports, etc. [10]

I.7.2 Robotique :

- Permet aux machines de comprendre les actions humaines afin d'interagir de manière optimale avec leur environnement.
- Robots d'assistance personnelle (ex : discerner qu'une personne tend un objet). [11]

I.7.3 Réalité augmentée / réalité virtuelle (AR/VR) :

- Amélioration de l'expérience utilisateur dans des mondes virtuels.
- Suivi des gestes pour des expériences immersives. [12]

I.7.4 Santé et assistance aux personnes âgées :

- Suivi des activités quotidiennes (ADL).
- Détection de chutes ou de comportements atypiques. [11]

I.7.5 Commerce et étude comportementale :

- Analyse des comportements des consommateurs dans les magasins (ex : sélectionner un produit).
- Publicité contextuelle ou interactive en fonction des actions des utilisateurs. [13]

I.7.6 Conduite autonome / observation des conducteurs :

- Reconnaissance des gestes du conducteur (ex : tenir le volant, utiliser un téléphone).
- Surveillance de l'attention et des interactions avec les commandes du véhicule. [12]

I.7.7 Jeux vidéo / interfaces utilisateur :

- Contrôle des jeux par le biais de gestes.
- Interfaces naturelles basées sur des actions physiques. [14]

I.8 Conclusion :

La reconnaissance des interactions entre humains et objets (Human-Object Interaction, HOI) représente un secteur fondamental de la vision informatique, destiné à saisir non seulement la présence d'individus et d'objets dans un cadre, mais également la nature des interactions qui s'y déroulent. Cette mission enrichit l'analyse visuelle par une compréhension contextuelle approfondie, qui est cruciale pour des domaines tels que la robotique, la surveillance vidéo intelligente ou la réalité augmentée. Malgré les avancées significatives grâce à des méthodes reposant sur l'apprentissage profond, des obstacles demeurent, notamment en ce qui concerne la variabilité des postures, les occlusions, et l'interprétation précise des actions. Les recherches à venir visent des modèles plus résilients, capables d'analyser des scènes complexes en temps réel avec une meilleure capacité de généralisation.

Chapitre II : Les méthodes de classification

II.1 Introduction :

Ces dernières années, la recherche sur les modèles de détection d'objets a suscité beaucoup d'attention en raison de l'essor du marché de la vision par ordinateur. Pour interpréter une image ou une vidéo, l'ordinateur doit d'abord détecter les objets et également estimer précisément leur emplacement dans l'image/vidéo avant de les classer. La détection d'objets se compose de plusieurs sous-tâches telles que la détection de visage, la détection de piétons, la détection de squelettes, etc., et a des cas d'utilisation courants tels que les systèmes de surveillance et les voitures autonomes. Dans cet article, nous allons passer en revue quelques types différents d'algorithmes de détection d'objets qui sont populaires de nos jours. Il existe différents types d'algorithmes de détection d'objets, certains sont des techniques traditionnelles et d'autres sont des techniques modernes développées récemment. Ces architectures diffèrent les unes des autres en fonction de leur précision, de leur vitesse et des ressources matérielles requises.

II.2 Les méthodes de classification

II.2.1 Réseau de neurones convolutif basé région (R-CNN)

La méthode R-CNN (Region with Convolutional Neural Networks) est une approche puissante pour la détection d'objets dans les images. Elle combine des techniques de proposition de régions (région proposals) avec des réseaux de neurones convolutifs (CNN) pour identifier et localiser des objets dans une image. Sa structure se base sur un flux de travail en trois étapes. La première étape consiste à créer des propositions de régions à l'aide d'algorithmes tels que Selective Search, qui repèrent des zones d'intérêt dans une image, sans tenir compte des catégories d'objets. La deuxième étape inclut l'extraction de caractéristiques de chaque région proposée à l'aide d'un réseau neuronal convolutionnel approfondi. Ce réseau convertit chaque région en un vecteur de caractéristiques de taille constante. Enfin, dans la troisième étape, ces vecteurs sont analysés par des classificateurs SVM dédiés à chaque catégorie, complétés par un module de régression de boîte englobante qui affine avec précision la position des objets détectés. Cette méthode a permis de combiner les atouts des réseaux profonds avec une approche modulaire de détection, tout en capitalisant sur des propositions régionales pour alléger la complexité du problème. [15]

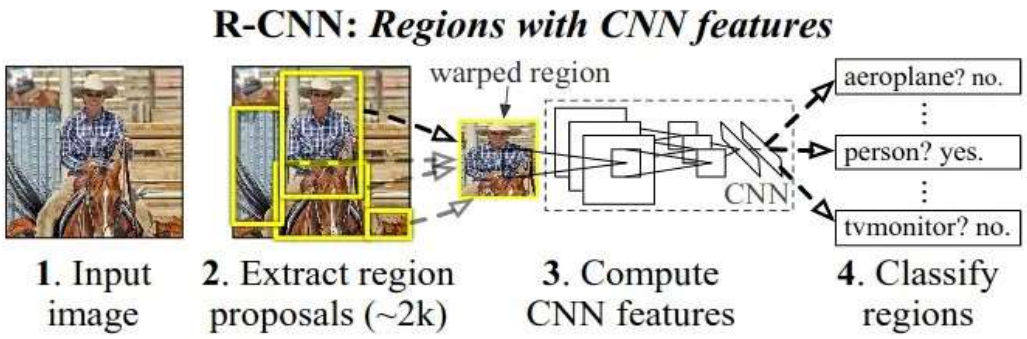

Figure II.1: Architecture de model R-CNN.

II.2.2 Réseau de neurones convolutif basé région rapide (Fast R-CNN)

Ross Girshick à proposer un nouvel algorithme d'apprentissage qui corrige les inconvénients du R-CNN et du SPPNet, tout en améliorant leur vitesse et leur précision. Nous appelons cette méthode le R-CNN rapide (Fast R-CNN) car elle est comparativement rapide à entraîner et à teste. Ce réseau prend en entrée une image entière et un ensemble de propositions d'objets. Le réseau traite d'abord l'image entière avec plusieurs couches convolutionnelles (conv) et de mise en commun maximale pour produire une carte de caractéristiques. Ensuite, pour chaque proposition d'objet, une couche de mise en commun des régions d'intérêt (RoI) extrait un vecteur de caractéristiques de longueur fixe de la carte de caractéristiques. Chaque vecteur de caractéristiques est introduit dans une séquence de couches entièrement connectées (FC) qui se ramifient finalement en deux couches de sortie sœurs :

✓ Une couche qui produit des estimations de probabilité softmax sur K classe d'objets plus une classe de "fond".

✓ Une couche qui produit quatre nombres à valeur réelle pour chacune des K classe d'objets. Chaque ensemble de 4 valeurs code les positions raffinées de la boîte de liaison pour l'une des K classes.

Le Fast R-CNN obtient un résultat de (66.1%) sur le VOC et le meilleur résultat sur le VOC avec un a mAP de 65,7 % (et 68,4 % avec des données supplémentaires). [16]

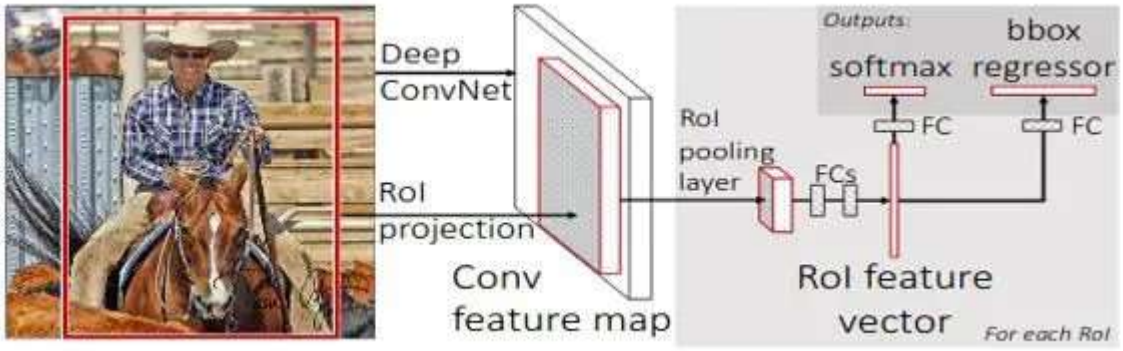

Figure II.2: Architecture de model Fast R-CNN.

II.2.3 Réseau de neurones convolutif basé région plus rapide (Faster R-CNN) :

Faster R-CNN, une architecture avant-gardiste pour la détection d'objets en temps réel. Cette technique intègre un Réseau de Propositions de Régions (RPN) qui partage les caractéristiques convolutives avec le réseau de détection, permettant ainsi de produire des propositions de régions de manière rapide et efficace. Le RPN est entraîné de façon continue pour prédire conjointement les contours des objets et les scores de pertinence à chaque emplacement. En associant le RPN avec le détecteur Fast R-CNN, le système global atteint des performances exceptionnelles sur des ensembles de données comme PASCAL VOC et MS COCO, tout en conservant une vitesse de traitement élevée. [17]

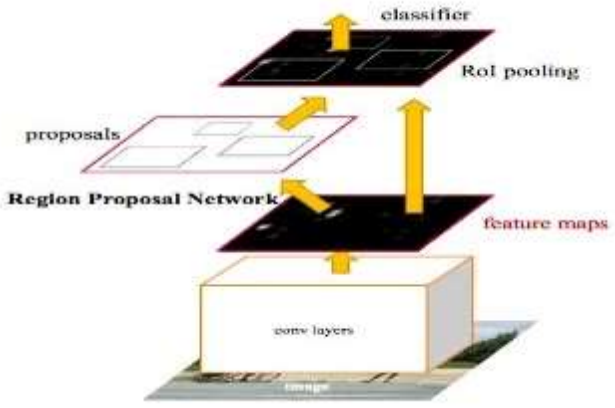

Figure II.3 : Architecture de model Faster R-CNN.

II.2.4 Réseau de neurones convolutif basé région mask (Mask R-CNN)

Kaiming He et al [17] ont proposés une autre architecture appelé Mask R-CNN, étendu de Faster R-CNN en ajoutant une branche pour la prédiction d'un masque d'objet en parallèle avec la branche existante pour la reconnaissance de la boîte englobante. Le Mask R-CNN masqué est conceptuellement simple : Le Faster R-CNN a deux sorties pour chaque objet candidat, une étiquette de classe et un décalage de la boîte englobante ; à cela, nous ajoutons une troisième branche qui produit le masque de l'objet.

La branche masque est un petit FCN (Fully Convolutional Network) appliqué à chaque RoI (Region of Interest), prédisant un masque de segmentation d'une manière pixel à pixel, ce principe d'alignement pixel à pixel c'est la pièce manquante du Fast/Faster R-CNN Le résultat obtenu sur la base de données MS COCO + fine est 36.4 mAP sur la validation en utilisant ResNet-50-FPN. [18]

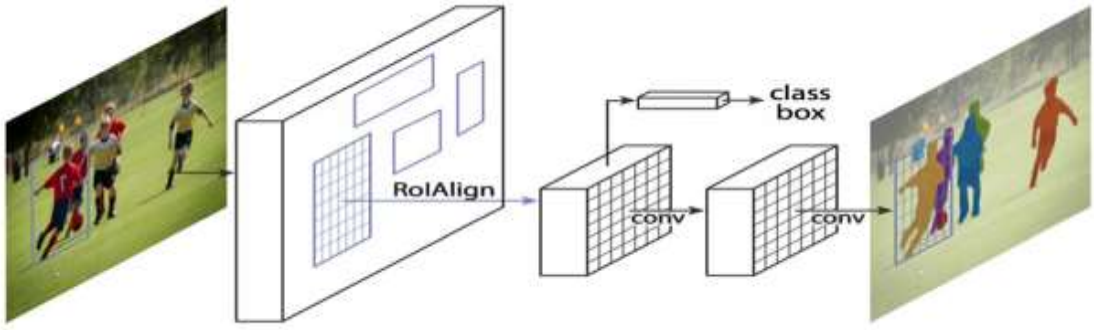

Figure II.4 : Architecture de model Mask R-CNN.

II.2.5 SSD: Single Shot MultiBox Detector

L'approche de la SSD est basée sur un réseau convolutif feed-forward qui produit une collection de boîtes englobantes de taille fixe et des scores pour la présence d'instances de classes d'objets dans ces boîtes, suivi d'une étape de suppression non maximale pour produire les détections finales Les premières couches du réseau sont basées sur une architecture standard utilisée pour la classification d'images de haute qualité (VGG-16) (tronquée avant toute couche de classification),

qui s'appelle réseau de base. Ensuite une structure auxiliaire au réseau pour produire des détections avec les caractéristiques clés suivantes :

✓ Cartes de caractéristiques multi-échelles pour la détection.

✓ Prédicteurs convolutifs pour la détection.

✓ Boîtes et aspect par défaut. [19]

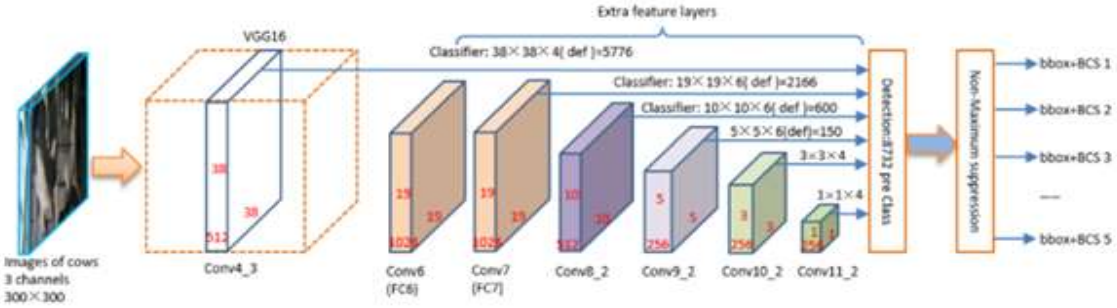

Figure II.5 : Architecture de model SSD.

II.2.6 YOLO: You Only Look Once

You Only Look Once ou YOLO est l'un des algorithmes populaires de détection d'objets utilisé par les chercheurs du monde entier. Il a été décrit pour la première fois dans en 2015 dans l'article de Joseph Redmon et al [20].

Le réseau utilise les caractéristiques de l'image entière pour prédire chaque boîte englobante. Il prédit également toutes les boîtes englobantes de toutes les classes d'une image simultanément. Cela signifie que ce réseau raisonne globalement sur l'ensemble de l'image et sur tous les objets qu'elle contient. La conception YOLO permet un apprentissage de bout en bout et des vitesses en temps réel tout en maintenant une précision moyenne élevée. [20]

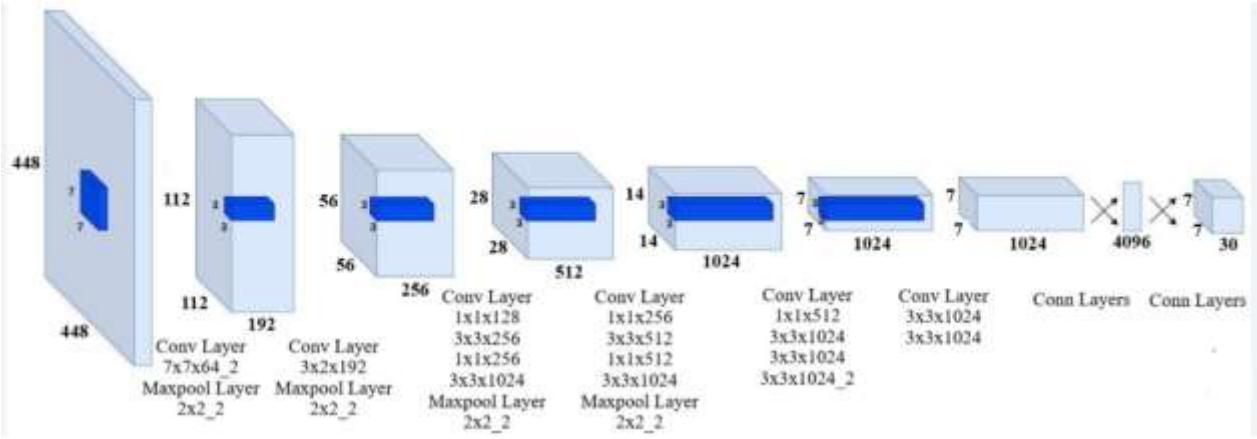

Figure II. 6 : Architecture de model YOLO.

II.2.7 U-net

U-Net est un modèle de réseau de neurones convolutifs, connue pour sa capacité à capturer des informations contextuelles à différentes échelles tout en conservant des détails spatiaux importants. Cette architecture est particulièrement adaptée à la segmentation sémantique des images.
L'architecture d'U-Net se compose de deux chemins principaux : le chemin contractuel (Encodeur) et Chemin d'Expansion (Décodeur), permettant de réduire puis de restaurer la résolution de l'image tout en intégrant les informations contextuelles à chaque étape, ce qui lui confère une architecture en forme de « U ».

• **Le chemin contractuel (encodeur)** : représente la partie gauche de l'architecture U-Net. Son rôle est de capturer les caractéristiques de l'image à différentes échelles en utilisant des couches de :

✓ **Convolutions** : Chaque étape du chemin contractant consiste en deux convolutions successives avec des filtres 3x3, suivies d'une fonction d'activation ReLU.

✓ **Max-Pooling** : Après les convolutions, une opération de max-pooling 2x2 est appliquée pour réduire la dimension spatiale de l'image tout en augmentant la profondeur des canaux. Cela permet d'extraire des caractéristiques de plus en plus complexes tout en réduisant la résolution spatiale.
Cette séquence de convolutions suivies d'un max-pooling se répète généralement 4 fois.

• **Le chemin (décodeur)** : correspond à la partie droite de l'architecture U-Net. Il est responsable de la localisation précise des caractéristiques grâces :

✓ **Transpositions Convolutionnelles (Up-sampling)** : Chaque étape du chemin expansif commence par une opération de transposition convolutionnelle 2x2 pour augmenter la dimension spatiale de l'image.

✓ **Convolutions :** Les convolutions 3x3 suivies d'une fonction d'activation ReLU sont appliquées pour affiner les caractéristiques et les détails.

• **Connexions de Pont (Skip Connections, Concatenate)** : Les connexions de pont relient les couches correspondantes du chemin contractant et du chemin expansif. Elles combinent les caractéristiques des deux chemins, permettant de conserver les détails tout en ajoutant un contexte global.

• **Couches de Sortie** : La couche finale est une convolution 1x1 qui réduit le nombre de canaux au nombre de classes de segmentation désiré, généralement suivie d'une fonction d'activation sigmoïde ou softmax selon le problème traité (segmentation binaire ou multi-classes).

Le modèle U-Net présente plusieurs avantages qui en font un choix populaire pour la segmentation d'images. [21]

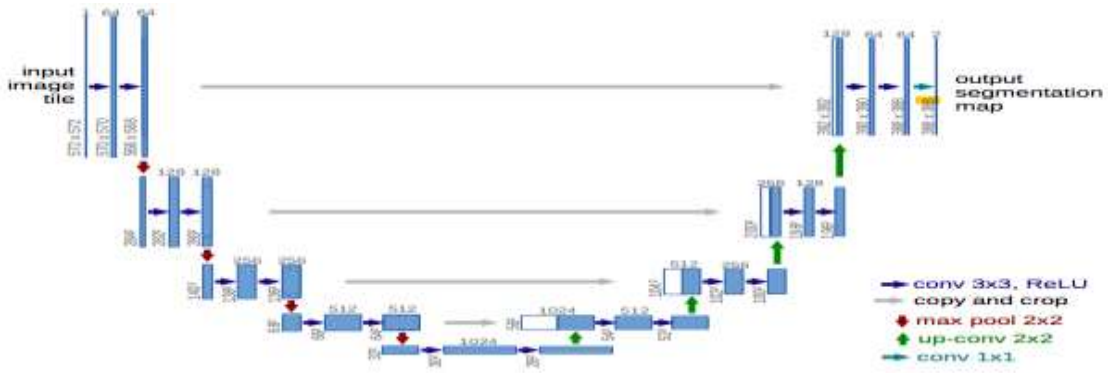

Figure II. 7 : Architecture U-Net.

II.2.8 U-Net++

Vise à améliorer la précision de la segmentation en incluant des couches de convolution denses directement entre l'encodeur et le décodeur. Dans l'image ci-dessous, la partie noire représente l'U-net traditionnel. La partie en vert représente les couches additionnelles. Le nombre de paramètres ainsi que le temps d'entrainement du réseau est nettement plus élevé. C'est la raison principale pour laquelle il n'y a pas d'architecture similaire avec des convolutions 3D. [22]

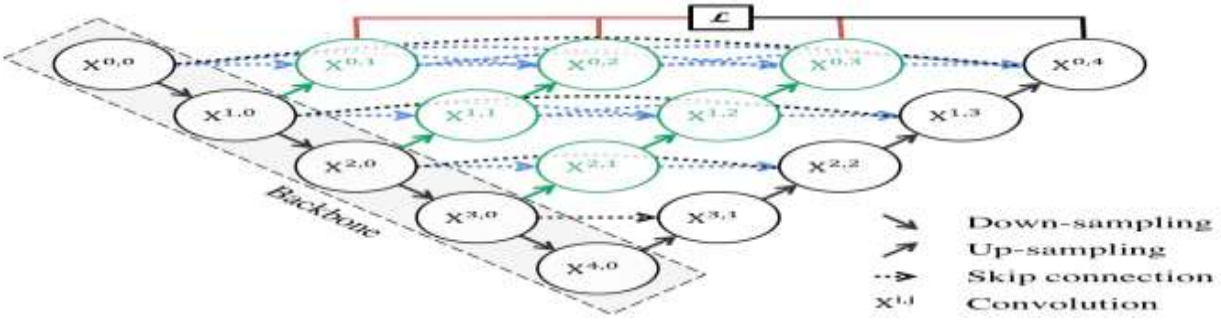

Figure II. 8: Architecture U-Net++.

II.2.9 LKC-Net

LKC-Net, ou Large Kernel Convolution Object Detection Network, est une nouvelle architecture conçue pour améliorer les performances de détection d'objets en exploitant les convolutions à noyau large. Cette architecture répond aux limitations associées aux petites convolutions à noyau, qui ont souvent du mal à capturer efficacement les caractéristiques sémantiques en raison de leurs champs réceptifs limités. [23]

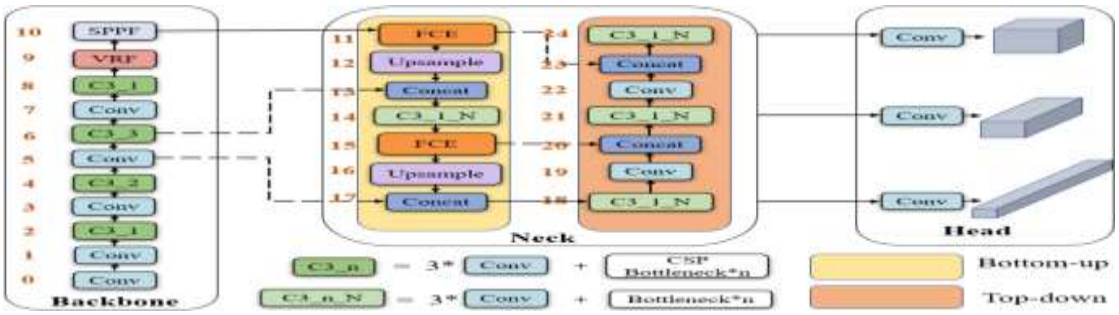

Figure II. 9: Structure LKC-Net.

II.3 Estimation de la pose humaine (OpenPose, HRnet)

II.3.1 OpenPose

À partir des nombreuses premières couches, la bibliothèque OpenPose extrait les caractéristiques d'une image. Les caractéristiques extraites sont ensuite intégrées à deux divisions parallèles de couches de réseau convolutif. La première division prédit un ensemble de 18 cartes de confiance, chacune représentant une partie spécifique du squelette de pose humaine. La branche suivante prédit un autre ensemble de 38 champs d'affinité de partie (PAF) indiquant la position d'association entre les corridors. Les étapes suivantes permettent de nettoyer les pronostics établis par les branches. À l'aide de graphiques de confiance, des graphes duaux sont créés entre les dyades

de corridors. Les valeurs de PDF et les liens faibles sont éliminés dans les graphes bipartis. En appliquant toutes les étapes données, les squelettes de pose humaine peuvent être estimés et attribués à chaque personne de l'image. [24]

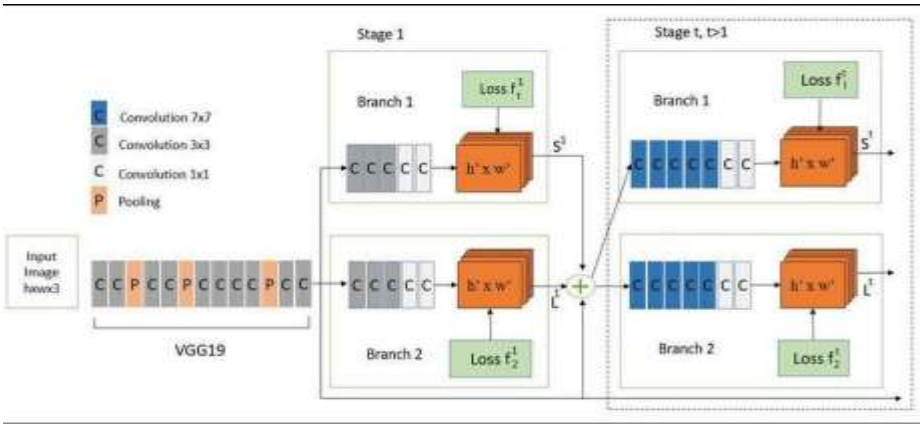

Figure II. 10 : Open Pose Architecture.

II.3.2 HRnet

Le réseau HRNet proposé maintien des représentations haute résolution tout au long du processus. Nous partons d'un flux de convolution haute résolution et ajoutons un à un des flux de convolution haute-basse résolution. Les flux multi résolution sont connectés en parallèle. Nous obtenons un réseau composé de plusieurs étapes (quatre dans la conception actuelle), et l'étape n contient n flux correspondant à n résolutions. Nous effectuons des fusions multi résolutions, échangeant les informations entre les flux parallèles à plusieurs reprises.

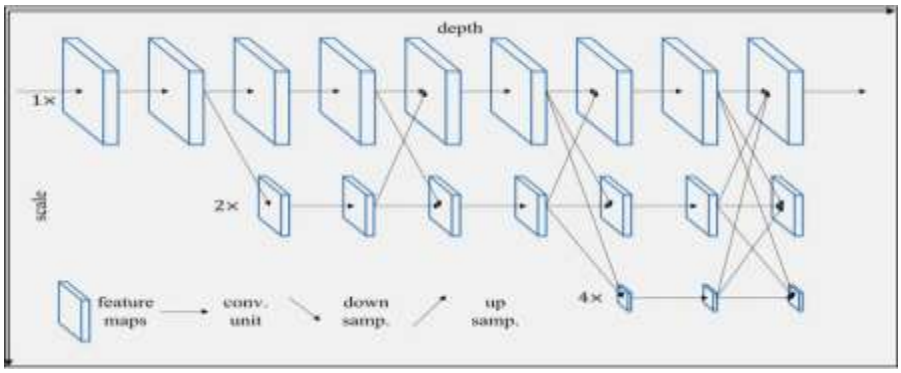

Figure II.11 : HRNet Architecture.

La force sémantique et la précision spatiale de la représentation haute résolution apprise par HRNet reposent sur deux aspects : premièrement, notre approche connecte les flux de convolution

de haute à basse résolution de manière parallèle plutôt que séquentielle. Par conséquent, contrairement à la récupération de la haute résolution à partir de la basse résolution, notre approche permet de maintenir directement la haute résolution, et la représentation apprise est donc spatialement plus précise. Alors que la plupart des schémas existants fusionnent les représentations haute résolution bas niveau et basse résolution haut niveau sur échantillonnées, nous proposons d'effectuer des fusions multi-résolutions répétitives pour améliorer les représentations de solutions plus élevées à l'aide des représentations basse résolution, et inversement. Ainsi, toutes les représentations de haute à basse résolution gagnent en sémantique. [24]

II.4 Fusion des caractéristiques pour la classification des interactions

La combinaison des caractéristiques pour la classification des interactions implique l'intégration d'informations tirées de diverses sources ou modalités afin d'optimiser la prise de décisions et d'améliorer l'efficacité des classifieurs [25]. Cette combinaison peut s'effectuer à divers niveaux :

- **Intégration des données :** Rassembler directement les données ou paramètres provenant de capteurs ou de méthodes d'extraction avant la classification.
- **Intégration des décisions :** Classifier indépendamment les données issues de différentes sources, puis amalgamer les résultats des classifieurs [25]. Un exemple sophistiqué est le cadre multi-échelle et multi-modal qui conjugue des interactions dans le domaine fréquentiel et des mécanismes d'attention croisée pour renforcer la représentation des caractéristiques, comme dans la classification d'images venant de perspectives multiples (ex. inspection par rayons X)[26]. Ce cadre comprend :

- Un module d'interaction dans le domaine fréquentiel pour saisir des détails précis.
- Un module d'attention multi-échelle pour accentuer les interactions entre perspectives.
- Un module de fusion par attention convolutionnelle qui précise et combine efficacement les caractéristiques tout en éliminant les redondances.

II.5 Architecture basées sur les transformes (HOI transformer)

Les premiers modèles de Deep learning se concentraient principalement sur les tâches de traitement du langage naturel (NLP) visant à amener les ordinateurs à comprendre le langage humain naturel et à y répondre. Ils ont deviné le mot suivant dans une séquence basée sur le mot précédent.

Pour mieux comprendre, pensez à la fonction de saisie semi-automatique de votre smartphone. Il fait des suggestions en fonction de la fréquence des paires de mots que vous tapez. Par exemple, si vous tapez souvent « Je vais bien », votre téléphone suggère automatiquement « bien » après que vous ayez tapé « Je vais ». [27]

II.6 Modèles par requêtes (QPIC, DETR-like)

II.6.1 QPIC (Query-Based Pairwise Interaction and Context-aware model)

QPIC est un modèle innovant conçu pour optimiser la détection des interactions entre les humains et les objets (Human-object Interaction Detection - HOI), qui s'inscrit dans le cadre plus vaste de la compréhension de scènes visuelles en vision par ordinateur [28].

✓ **Le terme QPIC signifies** : Interrogation basée, interaction par paire et contexte attentif

Cette création s'inspire profondément de l'architecture Transformer, notamment de DETR (Detection Transformer), afin d'identifier efficacement les interactions au sein d'une image. [28]

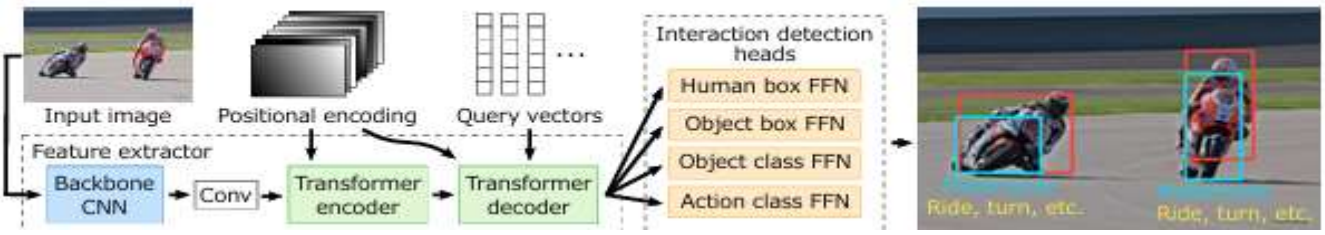

Figure II.12 : QPIC architecture.

II.6.2 DETR-like

L'architecture globale de DETR est étonnamment simple et est illustrée à la figure 13. Elle comprend trois composants principaux, décrits ci-dessous : une structure CNN pour extraire une représentation compacte des caractéristiques, un transformateur encodeur-décodeur et un réseau à réaction directe (FFN) simple qui effectue la prédiction de détection finale.

Contrairement à de nombreux détecteurs modernes, DETR peut être implémenté dans n'importe quel Framework d'apprentissage profond fournissant une structure CNN commune et une implémentation d'architecture de transformateur en quelques centaines de lignes seulement. Le code d'inférence de DETR peut être implémenté en moins de 50 lignes dans PyTorch [29]. Nous espérons que la simplicité de notre méthode attirera de nouveaux chercheurs dans la communauté de la détection.

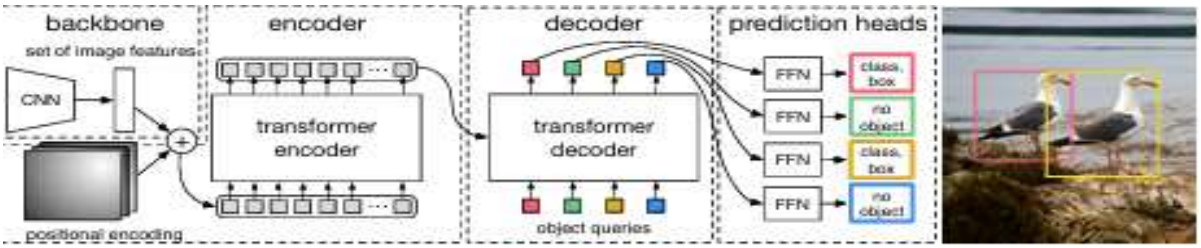

Figure II.13 : DETR-like architecture.

II.7 Modélisation des relations humain-objet avec des graphes

Chaque paire homme-objet peut avoir plusieurs étiquettes d'interaction et chaque scène peut inclure de nombreux humains et objets. Nous simplifions la tâche en exécutant un détecteur d'objets pré-entraîné qui détecte les humains et les objets dans une image. Détecter les interactions entre paires homme-objet est une tâche complexe. Des méthodes simples, comme extraire individuellement des caractéristiques de la localisation des humains et des objets et les analyser, sont inefficaces, car elles ignorent les informations contextuelles de l'environnement et les relations spatiales de la paire homme-objet. Des extensions, comme l'utilisation de boîtes d'union pour modéliser les relations spatiales/le contexte, sont également insuffisantes, car elles ne modélisent pas explicitement les interactions. Pour résoudre ces problèmes, nous proposons un réseau multi-branches avec des branches spécialisées. Le VSGNet proposé se compose de la branche visuelle qui extrait individuellement les caractéristiques visuelles des humains, des objets et du contexte environnant ; de la branche d'attention spatiale qui modélise les relations spatiales entre la paire homme-objet ; et de la branche convolutionnelle graphique qui considère la scène comme un graphe dont les humains et les objets sont les nœuds et modélise les interactions structurelles. L'architecture du modèle proposé avec les branches est illustrée dans la Fig. II.14. [30]

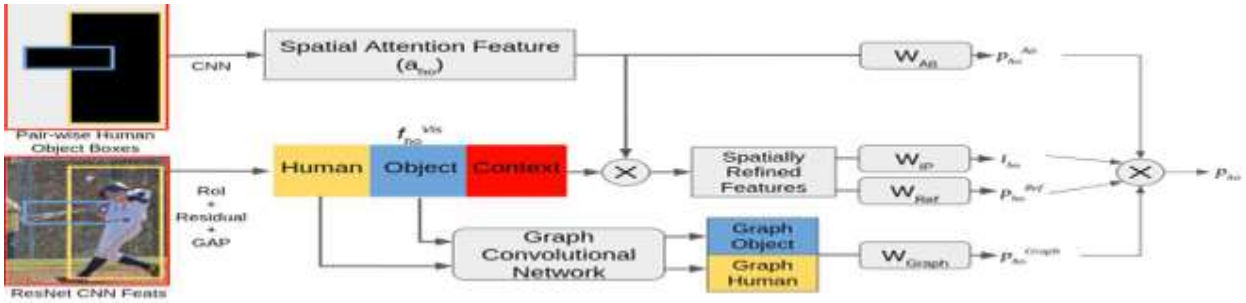

Figure II.14 : Model Architecture.

II.8 Conclusion

Dans ce chapitre, nous avons fourni une vue d'ensemble des approches de détection d'objets fondées sur l'apprentissage profond. L'accent a été mis sur les méthodes dérivées de l'architecture R-CNN, en analysant en détail leur pipeline de traitement, les variantes successives (Fast R-CNN, Faster R-CNN, Mask R-CNN) ainsi que leurs contributions respectives à l'amélioration des performances. Nous avons également passé en revue les méthodes récentes les plus performantes, en mettant en évidence leurs innovations architecturales et leurs résultats sur les principaux benchmarks.

Chapitre III : Conception et implémentation.

III.1 Introduction

Dans ce chapitre, nous allons explorer les multiples phases de la création et de l'exécution pour identifier divers objets (voiture, personne et moto, etc.) dans une série d'images à l'aide d'un langage approprié (dans notre cas, Python), ainsi que le matériel et les bibliothèques utilisées, les divers tests effectués et les résultats obtenus.

III.2 Conception :

III.2.1 Base des données : [31]

Figure III.1: Image sans bruit.

L'image sans bruit, illustrée dans la Figure III.1, représente une image nette de dimensions 300 x 168 pixels. Le fichier est de type JPG (.jpg) et sa taille réelle est de 11,4 Ko (11 706 octets), tandis que l'espace occupé sur le disque est de 12,0 Ko (12 288 octets). Cette image constitue une référence visuelle claire, utilisée comme base comparative pour évaluer les effets du bruit dans les étapes ultérieures de traitement.

Figure III. 2: Image bruitée (sombre).

L'image avec bruit, présentée dans la Figure III.2, montre une altération de l'image originale par l'ajout de bruit de type " sombre ", ce qui dégrade la qualité visuelle. Cette image conserve les mêmes dimensions que l'image nette, soit 300 x 168 pixels, et est également enregistrée sous le format JPG (.jpg). Sa taille réelle est de 9.15 Ko (9,370 octets), tandis que l'espace occupé sur le disque est de 12,00 Ko (12,288octets). Cette réduction de taille par rapport à l'image sans bruit reflète la perte d'informations due à la dégradation introduite par le bruit.

Figure III. 3: Image bruitée (claire).

L'image avec bruit de type "sel", illustrée dans la Figure III.3, résulte de l'ajout de points blancs parasites qui altèrent la netteté de l'image d'origine. Elle conserve une résolution de 300 x 168 pixels et est enregistrée au format JPG (.jpg). Sa taille réelle est de 14,6 Ko (15 011 octets), et elle occupe 16,0 Ko (16 384 octets) sur le disque. Cette augmentation de la taille, par rapport à l'image sans bruit, peut s'expliquer par la présence de pixels fortement contrastés qui complexifient la compression JPEG.

III.2.2 Choix des algorithmes

III.2.2.1 Yolov5

YOLOv5 (You Only Look Once version 5) est un modèle de réseau de neurones profonds élaboré pour la reconnaissance d'objets en temps réel dans des images ou des séquences vidéo. Développé par Ultralytics avec PyTorch, il se démarque par sa précision et sa simplicité d'utilisation. [32]

Figure III. 4: Image nette par la méthode yolov5.

Figure III. 5: Image bruitée (sombre) par la méthode yolov5.

Figure III. 6: Image bruitée (claire) par la méthode yolov5.

III.2.2.2 Faster R-CNN

Faster R-CNN (Faster Region-based Convolutional Neural Network) est un modèle sophistiqué de détection d'objets qui optimise les résultats de ses ancêtres (tels que Fast R-CNN) en incorporant un mécanisme de propositions régionales (RPN) directement au sein du réseau, rendant la détection plus rapide et plus précise. [33]

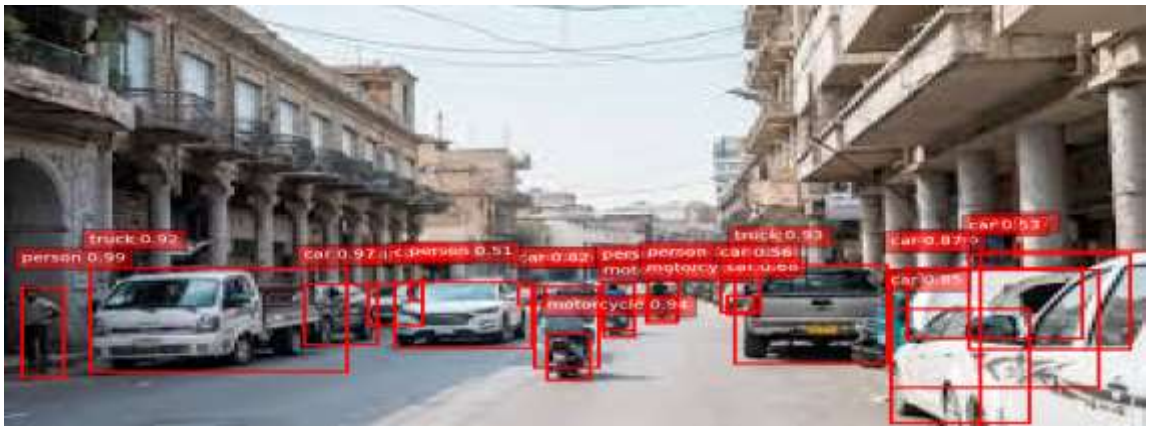

Figure III. 7: Image nette par la méthode Faster R-CNN.

Figure III. 8: Image bruitée (sombre) par la méthode Faster R-CNN.

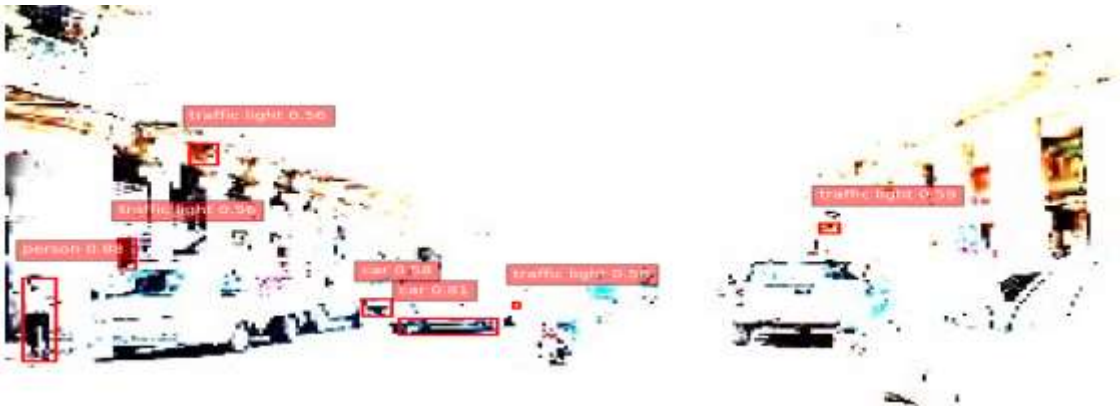

Figure III. 9: Image bruitée (claire) par la méthode Faster R-CNN.

III.2.2.3 SSD (Single Shot MultiBox Detector)

Le SSD est un modèle simple de détection d'objets capable de repérer avec précision les éléments présents sur une image. En une seule étape, il localise les limites de chaque objet représenté et détermine leur catégorie. Contrairement aux architectures telles que R-CNN qui nécessitent plusieurs étapes de traitement, le SSD parvient à cette performance de détection en une seule fois grâce à l'analyse multi-niveaux de son réseau neuronal. Ses boîtes englobantes et prédictions de classe sont le fruit direct de l'extraction de caractéristiques opérée à différentes échelles, ce qui lui permet d'identifier avec justesse aussi bien les plus petits détails que les plus grandes composantes de la scène visualisée. [34]

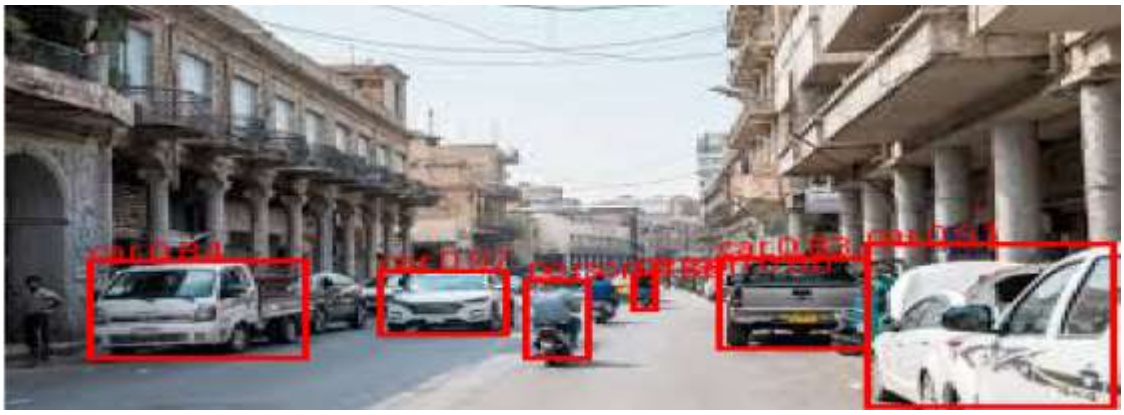

Figure III. 10: Image nette par la méthode SSD.

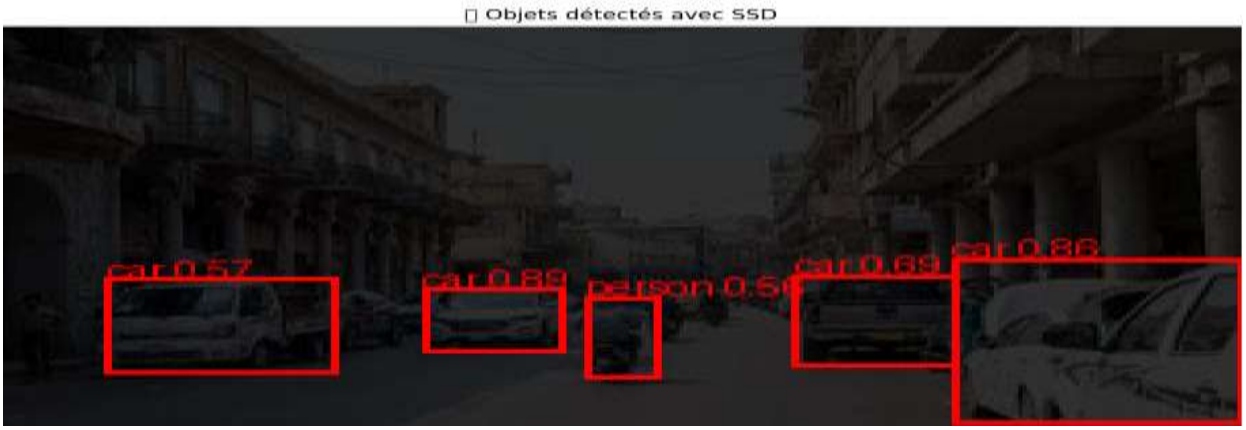

Figure III. 11: Image bruitée (sombre) par la méthode SSD.

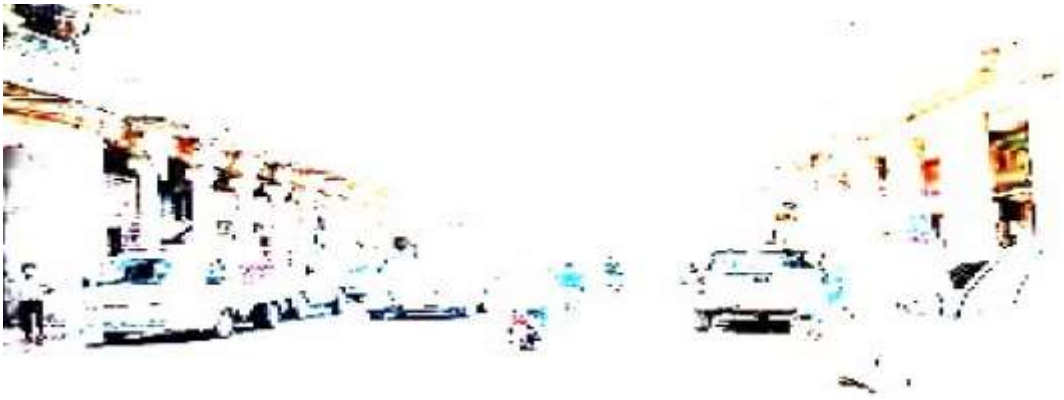

Figure III. 12: Image bruitée (claire) par la méthode SSD.

III.2.2.4 U-Net++

U-Net++ est une architecture de réseau de neurones convolutifs (CNN) basée sur le modèle encodeur-décodeur, dotée de liaisons denses et imbriquées entre ses différentes couches. C'est une tentative d'améliorer la segmentation en diminuant la différence sémantique entre les caractéristiques tirées à diverses profondeurs du réseau. [35]

Figure III. 13: Image nette par la méthode U-Net++.

Figure III. 14: Image bruitée (sombre) par la méthode U-Net++.

Figure III. 15 : Image bruitée (claire) en utilise la méthode U-Net++.

III.2.3 Les critères d'évaluations

VP : vrais positifs est le nombre d'instances positives correctement classifiées.

FP : faux positifs est le nombre d'instances négatives et qui sont prédites comme positives.

FN : faux négatifs est le nombre d'instances positives classifiées comme négatives.

VN : vrais négatifs est le nombre d'instances négatives correctement classifiées. À partir de la matrice de confusion on peut calculer plusieurs métriques.

III.2.3.1 Précision

La précision (précision) est le pourcentage de prédictions positives qui sont correctes. [36]

$$\text{Précision} = \frac{v_p}{v_p + f_p} \qquad (12)$$

III.2.3.2 Recall

La sensitivité ou le rappel (Recall) est le pourcentage des instances positives correctement identifiées. [37]

$$\text{Recall} = \frac{v_p}{v_p + f_n} \quad (13)$$

III.2.3.3 F1-score:

Le score F1 représente la moyenne harmonique de la précision et du rappel. C'est une mesure unique équilibrée qui prend en compte l'importance des deux métriques. [38]

$$\text{F1-score} = 2 \times \frac{précision \times Recall}{précision + Recall} \quad (14)$$

III.3. Implémentation et résultats

III.3.1 Environnement matériel

L'environnement matériel consiste en un ordinateur DELL ayant les caractéristiques suivantes :

- ✓ Usage : Générale
- ✓ Mémoire Vive : 8 GO
- ✓ Disque Dure : 500 GO
- ✓ Taille Ecran : 14,1
- ✓ Processeur: Core i5 Intel
- ✓ Graphics: Intel
- ✓ Windows: 7

III.3.2 Environnement logiciel

III.3.2.1 Langage Python

C'est un langage de programmation de haut niveau interprété (il n'y a pas d'étape de compilation), interactif et orienté objet avec une sémantique dynamique. Python est conçu pour être hautement lisible et utilise fréquemment des mots clés en anglais, alors que d'autres langages utilisent la ponctuation, il a moins de constructions syntaxiques. Il est très sollicité par une large communauté de développeurs et de programmeurs. On a utilisé la version 3.12. [39]

III.3.2.2 Bibliothèques utilisées :

✓ **Torch** : est une bibliothèque d'apprentissage automatique open source, un cadre de calcul scientifique et un langage de script basé sur le langage de programmation Lua. Il fournit une large

gamme d'algorithmes pour l'apprentissage en profondeur et utilise le langage de script LuaJIT et une implémentation en C sous-jacente. On a utilisé la version 2.7.0. [40]

✓ **Matplotlib** : Est une bibliothèque complète pour créer des visualisations statiques, animées et interactives en Python, il offre une alternative open source viable à MATLAB. On a utilisé les versions 3.10.1 et 3.10.3. [41]

✓ **Torchvision** : Le package torchvision se compose d'ensembles de données populaires, d'architectures de modèles et de transformations d'images courantes pour la vision par ordinateur. On a utilisé la version 0.22.0. [42]

✓ **Open CV :** est une vaste bibliothèque open source dédiée à la vision par ordinateur, à l'apprentissage automatique et au traitement d'images. Elle prend en charge une grande variété de langages de programmation tels que Python, C++, Java, etc. Elle peut traiter des images et des vidéos pour identifier des objets, des visages, voire l'écriture manuscrite. On a utilisé la version 4.11.0.86. [43]

✓ **NumPy :** NumPy est l'abréviation de « Numerical Python » et c'est un ensemble fondamental pour le calcul scientifique en Python. NumPy fournit à Python une vaste bibliothèque mathématique capable d'effectuer des calculs numériques de manière efficace et efficiente afin de pouvoir travailler avec des tableaux multidimensionnels et des structures de données matricielles, très courante dans les domaines de la science des données et de l'apprentissage automatique. On a utilisé les versions 2.1.3 et 2.2.5. [44]

III.3.3 Résultats d'entrainement

III.3.3.1 Image sans bruit

critères / Methods	Précision	Recall	F1-score
YOLO	0.14	0.33	0.20
Faster R-CNN	0.25	0.50	0.33
SSD	0.50	0.50	0.50
U-net++	0.23	0.21	0.22

Tableau III 1: Rapport de classification pour déférent des méthodes pour une image nette.

Pour une image nette, la méthode SSD offre les meilleures performances globales avec un F1-score de 0.50, surpassant les autres méthodes. Faster R-CNN présente un bon rappel mais une

précision plus faible, tandis que YOLO et U-net++ affichent des performances globalement faibles, indiquant une moindre efficacité pour cette tâche spécifique.

III.3.3.2 Image avec bruit (sombre) :

critères / Methods	Précision	Recall	F1-score
YOLO	0.14	0.33	0.20
Faster R-CNN	0.33	0,50	0,40
SSD	0.50	0.50	0.50
U-net++	0.18	1.00	0.31

Tableau III 2: Rapport de classification pour déférent des méthodes pour image avec bruit (sombre)

Dans des images sombres et bruitées, Parmi les modèles comparés, SSD se révèle le plus performant grâce à un bon compromis entre précision et rappel. Faster R-CNN offre une détection correcte avec une précision modérée. U-Net++ garantit une détection complète des objets, mais au prix d'un grand nombre de faux positifs. YOLO, en revanche, montre des résultats faibles sur l'ensemble des métriques. Ainsi, le choix du modèle dépendra des exigences spécifiques de l'application, notamment entre fiabilité globale et sensibilité maximale.

III.3.3.3 Image avec bruit (claire)

critères / Methods	Précision	Recall	F1-score
YOLO	0.14	0.33	0.20
Faster R-CNN	0.33	0.50	0.40
SSD	0	0	0
U-net++	0.19	0.75	0.31

Tableau III 3: Rapport de classification pour déférent des méthodes pour image avec bruit (claire)

Dans le cas des images avec bruit de type « claire », les performances des méthodes varient fortement. Faster R-CNN obtient les meilleurs résultats globaux (F1-score de 0.40), montrant une bonne robustesse face au bruit. U-Net++ détecte la majorité des objets (rappel élevé de 0.75) mais

avec beaucoup de faux positifs. YOLO montre une faible précision (0.14), tandis que SSD échoue complètement. Ces résultats soulignent l'importance de choisir une méthode adaptée au bruit ou d'intégrer un prétraitement de débruitage.

III.3.4 Comparaison

1. **SSD** :

- Robuste pour images nettes et bruitées sombres.
- Échec total en présence de bruit clair.
- Bon compromis précision / rappel.
- Recommandé pour environnements modérément bruités.

2. **Faster R-CNN** :

- Performances stables et solides dans tous les scénarios.
- Meilleur choix en présence de bruit clair.
- Moins précis que SSD mais plus constant.

3. **U-net++** :

- Très sensible (rappel élevé), surtout en cas de bruit.
- Faible précision ⇒ production de nombreux faux positifs.
- Adapté si détection maximale est prioritaire.

4. **YOLO** :

- Faible performance dans tous les cas.
- Rapide mais non adapté à ce contexte spécifique, probablement à cause de bruit ou de manque d'optimisation

Méthode	Robustesse au bruit	Points forts	Points faibles
SSD	Excellente sur image nette et bruit sombre Échec sur bruit clair	- Meilleur F1 -score global (0.50) - Bon équilibre précision/rappel	- Sensible au bruit clair - Pas de détection du tout dans ce cas
Faster R-CNN	Bonne robustesse dans toutes les conditions	- Stabilité des performances - Meilleur F1-score sur bruit clair	- Précision moyenne - Moins rapide que YOLO ou SSD

Méthode	Robustesse au bruit	Points forts	Points faibles
U-Net++	Variable : très bon rappel mais faible précision	- Très haute sensibilité (rappel jusqu'à 1.00)	- Génère beaucoup de faux positifs - Faible précision globale
YOLO	Faible robustesse dans tous les cas	- Traitement très rapide (en général)	- F1-score le plus bas - Faible performance sous bruit et sans bruit

Tableau III 4: Comparaison les résultats des méthodes.

III.4 Conclusion

Ce chapitre a été structuré en deux sections complémentaires. La première partie a porté sur la description de la base de données utilisée ainsi que sur la phase de conception du système de détection. Dans ce cadre, nous avons examiné en détail les architectures des modèles sélectionnés, à savoir YOLOv5, Faster R-CNN, SSD et U-Net++. Chacun de ces modèles a été analysé en termes de structure, de mécanisme de fonctionnement, de performances théoriques, et de pertinence par rapport aux exigences de la tâche de détection d'objets.

La seconde partie a été dédiée à l'implémentation pratique. Nous y avons décrit l'environnement de développement, les bibliothèques logicielles mobilisées (telles que PyTorch, OpenCV, NumPy, etc.) en précisant leur rôle dans le processus de traitement et d'apprentissage. Enfin, nous avons présenté les résultats expérimentaux obtenus, illustrant la capacité des modèles à détecter efficacement des objets dans des images, et mettant en évidence les différences de performances entre les approches.

Conclusion générale

Conclusion Générale et Perspectives

Ce mémoire a mis en lumière l'importance croissante de la détection des interactions humain-objet (Human-Object Interaction, HOI) dans le domaine de la vision par ordinateur, soulignant les avancées substantielles rendues possibles par les techniques d'apprentissage profond. À travers l'analyse comparative des architectures de réseaux de neurones, nous avons démontré leur capacité à accroître significativement la précision, la robustesse et la scalabilité des systèmes de détection. Ces architectures permettent notamment de mieux gérer des problématiques complexes telles que les occlusions, les interactions multiples simultanées, les variations contextuelles et les environnements non structurés.

Les résultats expérimentaux obtenus confirment le potentiel des approches étudiées pour un large éventail d'applications, notamment en sécurité intelligente, robotique collaborative, systèmes d'assistance à la personne, et interfaces homme-machine basées sur la compréhension des gestes et comportements.

Plusieurs perspectives de recherche se dessinent à l'issue de ce travail :

- Amélioration de la robustesse contextuelle : L'intégration de modules d'attention contextuelle ou de modélisation spatiale-temporelle pourrait renforcer la capacité des modèles à interpréter des scènes complexes dans des environnements dynamiques.
- Réduction de la dépendance aux annotations : L'exploration de méthodes d'apprentissage faiblement supervisé ou auto-supervisé pourrait permettre de tirer parti de larges volumes de données non annotées, tout en maintenant des performances compétitives.
- Fusion multi-modale : La combinaison de données visuelles avec d'autres modalités (profondeur, audio, capteurs inertiels) constitue une voie prometteuse pour améliorer la compréhension des interactions en environnements réels.
- Optimisation temps réel : Adapter les architectures pour un déploiement sur des dispositifs embarqués ou contraints en ressources (edge computing) est crucial pour les applications industrielles et mobiles.
- Compréhension sémantique plus fine : Étendre les modèles pour qu'ils reconnaissent non seulement les interactions, mais aussi leur intention ou leur finalité, pourrait ouvrir la voie à des systèmes capables d'anticiper les actions humaines dans un cadre prédictif.

Références:

[1] Bogusław Cyganek, Object Detection and Recognition in Digital Images: Theory and Practice, Wiley, 2013.

[2] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436–444. https://doi.org/10.1038/nature14539

[3] Kemal Oksuz, Baris Can Cam, Sinan Kalkan, and Emre Akbas, « Imbalance Problems in Object Detection : A Review », IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, n° 10, 2021, pages 3388-3415, https://ieeexplore.ieee.org/document/9042296 (lien externe à ibm.com).

[4] H.YEDJOUR. Détection de contours et suivi d'objet dans une séquence d'images par les réseaux de neurones impulsionnels Université Mohamed Boudiaf (USTO) ORAN Thèse Magister 2010.P13,14,P16,17,18,19,20,21,22 ;23,P24.

[5] Cao Tien Dung – Promotion 11, Institut de la Francophonie pour l'Informatique, " La vidéos surveillance ", Janvier 2007

[6] HORAUD, Radu, MONGA, Olivier, "Vision par ordinateur", 1995.

[7] Centre de recherche informatique de Montréal, "La vidéosurveillance intelligente", Avril 2009.

[8] Sung Wook Seol, Jee Hye Jang, Hyo Sung Kim, Chul Hun Lee, and Ki Gon Nam. An Automatic "Detection and Tracking System of Moving Objects Using Double Difference based Motion Estimation", 2003.

[9] Yiğithan Dedeoğlu. "Moving Object detection, Tracking and Classification for Smart Video Surveillance". August2004.

[10] Chao, Y.-W., Liu, Y., Liu, X., Zeng, H., & Deng, J. Learning to Detect Human-Object Interactions. (2018).

[11] GAO, C., & Nevatia, R. iHOI: Interact as Human with Human-Object Interaction Detection. (2018).

[12] Gupta, S., Malik, J., & Hebert, M. Visual Semantic Role Labeling: A Benchmark. (2015).

[13] Zhou, P., Zhan, W., & Tomizuka, M. HOI-aware Object Detection. (2019)

[14] The HICO-DET dataset Un des jeux de données les plus utilisés pour l'entraînement et l'évaluation des modèles HOI. Site « https://arxiv.org/abs/2305.09948

[15] R. Girshick et autre « Rich feature hierarchies for accurate object detection and semantic segmentation » IEEE Conference on Computer Vision and Pattern Recognition, 2014

[16] R. Girshick « Fast R-CNN », Conférence internationale de l'IEEE sur la vision par ordinateur, 2015

[17] S. Ren, K. He, R. Girshick, and J. Sun. « Faster R-CNN: Towards real-time object detection with region proposal networks ». Dans Conférence NIPS, 2015.

[18] K. He, G. Gkioxari, P. Doll´ ar, and R. Girshick « Mask R-CNN» dans les Acts Conférence internationale de l'IEEE sur la vision par ordinateur ICCV, 24 Janvier 2018

[19] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Fu, C. et Berg, A. C, «SSD: Single Shot MultiBox Detector » Preprint sur https://arxiv.org/abs/1512.02325, 2016

[20] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. «You only look once: Unified, real time object detection ». Preprint sur arXiv : 1506.02640, 2015

[21] R.BAROUDI, Z.BENCHIHA «Approche d'apprentissage profond pour la segmentation sémantique d'images», projet fin d'étude, Université d'Ain Temouchent - Belhadj Bouchaib, 24.06. 2024

[22] Site web « blent.a », https://blent.ai/blog/a/unet-computer-vision Consulté le 20/06/2024.

[23] Wang, W., Li, S., Shao, J. *et al.* LKC-Net : réseau de détection d'objets à convolution à noyau large. Sci Rep 13, 9535 (2023). https://doi.org/10.1038/s41598-023-36724-x

[24] R.Santhosh, V.Ganga «POSE ESTIMATION TECHNIQUES USING OPENPOSE, POSENET AND HRNET» Master Of Technology In Computer Science, Department Of Computer Science And Engineering, JNTUH University College Of Engineering Science And Technology, Hyderabad, Telangana, India. DOI : https://www.doi.org/10.56726/IRJMETS61308.

[25] Site web https://www.ensta-bretagne.fr/fr/fusion-dinformations-pour-la-classification

[26] S.Hong[1], Y.Zhou[2], W.Xu[1], Un cadre de fusion de fonctionnalités multi-échelles intégrant le domaine fréquentiel et l'attention croisée pour les inspections de sécurité par rayons X à double vue, [1]École d'automatisation, Université de technologie du Guangdong, Guangzhou 510006, Chine,[2]Institut des sciences et technologies de Guangzhou, Guangzhou 510006, Chinepanda@pandacow.cn, {zhouyanzhou, wcxu}@gdut.edu.cn.

[27] ArXiv advantage in transmers for robotic application 13 déc 2024 https://arxiv.org/html/2412.10599v1

[28] Tamura, A., & Sugimoto, A. (2021).QPIC: Query-Based Pairwise Interaction and Context-Aware Model for HOI Detection**.** CVPR 2021.

[29] N.Carion, F.Massa, G.Synnaeve, N.Usunier, A.Kirillov and S.Zagoruyko, End-to-End Object Detection with Transformers, arXiv: 2005.12872v3 [cs.CV] 28 May 2020.

[30] O.Ulutan, ASMIftekhar, B. S. Manjunath, VSGNet: Spatial Attention Network for Detecting Human Object Interactions Using Graph Convolutions, Department of Electrical and Computer Engineering, University of California, Santa Barbara, CA.

[31] Site web « https://www.irfaasawtak.com/iraq/2023/08/12 ».

[32] Site web « https://github.com/ultralytics/yolov5 ».

[33] S.Ren, K.He, R.Girshick, J.Sun, R-CNN plus rapide : vers une détection d'objets en temps réel avec des réseaux de proposition de région, arXiv : 1506.01497, 6 janvier 2016.

[34] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). SSD: Single Shot MultiBox Detector. In European Conference on Computer Vision (ECCV).

[35] Zhou, Z., Siddiquee, M. M. R., Tajbakhsh, and N., & Liang, J. (2018) .UNet++: A Nested U-Net Architecture for Medical Image Segmentation, ArXiv: 1807.10165.

[36] FATOUMATA Y & AMOR A. (2021), « Machine learning pour la maintenance prédictive », Mémoire de fin d'études, Université Larbi Ben Mhidi d'Oum El-Bouaghi, juillet.

[37] MIFDAL R, « Application des techniques d'apprentissage automatique pour la prédiction de la tendance des titres financiers », L'obtention De La Maitrise, Sous la direction de M. Edmond Miresco, École De Technologie Supérieure Université Du Québec, 2019.

[38] Saito & Rehmsmeier, The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets, PLOS ONE, 2015.

[39] OUNISSI M & HARNANE Z. (2020), « Modélisation et classification avec Deep Learning Application à la détection du Coronavirus Covid-19 », Mémoire de fin d'études, Université Mohamed Larbi Ben M'hidi - Oum El Bouaghi.

[40] GHEDIRI N & SAKRI Z. (2021), « ETUDE ET EXPERIMENTATION DES RESEAUX RESNET-50 ET INCEPTION-V3 DANS LA CLASSIFICATION DE CANCER DE LA PEAU », Mémoire de fin d'études, Université Larbi Ben M'Hidi d'Oum El Bouaghi.

[41] ZERARGUI F & BENZAOUI O. (2021), « Le discours de haine sur le web et les médias sociaux », Mémoire de fin d'études, Université de Bordj Bou Arreridj Mohammed El Bachir El Ibrahimi.

[42] Site web « https://github.com/pytorch/vision/blob/main/README.md ».

[43] Site web « https://www.geeksforgeeks.org/opencv-python-tutorial/ ».

[44] BELLAHMER H. (2020), « Implémentation et évaluation d'un modèle d'apprentissage automatique pour l'estimation de la valeur marchande de propriétés immobilières », Mémoire de fin d'études, UNIVERSITÉ MOULOUD MAMMERI DE TIZI-OUZOU.