

جامعة سيدي محمد بن عبد الله بفاس
+0800441 0444 600000 01 4044400 | 300
UNIVERSITÉ SIDI MOHAMED BEN ABDELLAH DE FES



كلية العلوم والتقنيات فاس
+05350011 1000011 1 0144441 - 300
FACULTÉ DES SCIENCES ET TECHNIQUES DE FÈS

DEPARTEMENT DE GENIE ELECTRIQUE

FILIERE : LICENCE SCIENCES ET TECHNIQUE EN GENIE ELECTRIQUE (LST GE)

LOGO SOCIETE



PROJET DE FIN D'ETUDES

PRÉPARÉ PAR

ACHRAF HANINE

POUR L'OBTENTION DU

DIPLOME LICENCE SCIENCES ET TECHNIQUE EN GENIE ELECTRIQUE

**DETECTION DE CAS COVID ET LA MESURE DE
LA TEMPERATURE.**

ENCADRÉ PAR :

M. HADDOU BELAYDI (CHU)

PR. JORIO Mohammed (FSTF)

SOUTENU LE JUILLET 2021, DEVANT LE JURY COMPOSÉ DE :

PR. JORIO Mohammed

FACULTE DES SCIENCES ET TECHNIQUES DE FES

PR. ESSBAI Najia

FACULTE DES SCIENCES ET TECHNIQUES DE FES

Remerciements

Je tiens à exprimer ma profonde gratitude à Monsieur le Directeur du centre hospitalier universitaire de Fès (CHU) de m'avoir donné l'occasion de découvrir le domaine de travail afin de le confronter et d'approfondir mes connaissances.

Je remercie vivement tous les agents travaillant dans l'équipe d'entretien de CHU qui se sont montrés dès le premier jour accueillants, chaleureux et disponibles.

Je remercie mon encadrant en entreprise M: HADDOU BELAYDI pour son encadrement et le temps qu'il m'a consacré. Je suis très reconnaissant pour tous les conseils qu'il a pu me prodiguer durant toute la durée de mon stage.

J'ai également le grand plaisir de remercier mon encadrant académique M JORIO Mohammed, pour son temps consacré, son aide et ses conseils. Je remercie tous les professeurs de la faculté des sciences et techniques surtout notre coordonnatrice Mme ESSBAI Najia, pour leur compréhension, leur disponibilité et leur soutien qu'ils nous ont apportés afin de nous assurer une formation solide.

Résumé :

Notre stage de fin étude s'est déroulé au sein de centre hospitalier universitaire de Fès.
(CHU)

Ce stage a pour but la détection d'une caméra numérique pour une unité de surveillance à base de l'intelligence artificielle, en effectuant l'étude de traitement sur la langage Python, Google colab et les algorithmes nécessaire Yolo, Yolocat.....

Le présent rapport élabore la conception et l'innovation d'une technologie qui va contrôler les gens dans un ascenseur soit de point de vue de protection ou bien de la surveillance

L'objectif est de passer en revue par quatre cardinales de fonctionnement de Smart Ascenseur Successivement :

Détection du corps physique, ensuite les algorithmes utilisé pour cette détection surtout la détection de la bavette, mesurage de la température enfin le projet smart ascenseur qui va collecter tous ces options pour permettra à CHU de gagner du temps à la protection de l'équipe ou les visiteurs pour éradiquer le virus.

Ce stage nous a permis de travailler en équipe, d'acquérir une expérience professionnelle dans une entreprise multinationale. En effet nous avons mis en pratique les connaissances acquises durant les trois ans d'études.

Tables de matière

Remerciements	2
Résumé :	3
Table de figure :.....	6
Liste des Abréviations :.....	8
Introduction général.....	9
Chapitre 1 : présentation du contexte générale	9
1.Présentation générale de centre hospitalier universitaire de Fès (CHU)	10
2.historique	10
3. Introduction de l'intelligence artificielle	10
3-1. L'intelligence artificielle (IA)	10
3-2 Machine Learning	11
3-3 Méthode d'apprentissage automatique	11
3-2-1.Apprentissage automatique supervisé : Supervised ML	11
3-3-2.Apprentissage automatique non supervisé : Unsupervised ML	13
3-3-3.Apprentissage automatique de renforcement : Reinforcement ML	13
3-4. Deep Learning : DL	14
Chapitre 2 : la détection des personnes et la surveillance dans CHU	14
1 Le principe de fonctionnement	14
1-1.Le rôle du Deep Learning dans la détection des objets	15
2.Exemples des algorithmes de détection et de classification utilisés	20
2-1.Génération du Bounding Box	20
2-2.YOLO (YOU ONLY LOOK ONCE : Vous ne regardez qu'une seule fois)	21
2-3.SSD (Single Shot MultiBox Detector) :	22
2.4.R-CNN	23
2.5. Yolact :	23
3.La surveillance dans CHU avec IA ((Yolo v5 , SSD avec mobile net ,Yolcat)	24
Chapitre 3 :la détection de la bavette et mesure de la Température des gens avec IA.	28
1.la détection du masque :	28
1-1 la famille YOLO en détection de masque et pourquoi Yolov4	28
1-2 Les étapes des algorithmes :	29
1-3 la description de démarches de programme dans la machine :	30
2.la mesure de la température :	36
2-1 caméra thermique :	36

Figure 33:image d une photo de caméra thermique	37
2-1-1 Détection des zones froides /chaudes.	37
2-3 capteur thermique :	37
3.Caméra thermique comme projet théorique	38
Chapitre 4 : Smart Ascenseur	39
1.cahier des charges	39
1-1 bête a cornes :.....	40
1-2Expression fonctionnelle du besoin :	41
1-3 Les normes :	41
2 Le système Raspberry :.....	43
2-1 La série de Raspberry utilisé dans le projet :	43
2-2 la fiche technique de caméra utilisée	44
3 Raspberry Pi 4 au service de Smart Ascenseur	45
4-Smart Ascenseur au service de CHU	46
Les solutions d entretien des ascenseurs :.....	47
Conclusion générale.....	48
Références.....	49

Table de figure :

Figure 1	illustration du fonctionnement du ML.
Figure 2	un exemple d'un problème de régression.
Figure 3	exemple d'un problème de classification
Figure 4	Les types du ML.
Figure 5	image qui illustre le résultat d'une détection d'objets
Figure 6	réseau de neurones
Figure 7	séparation de l'input par le réseau
Figure 8	CNN layers
Figure 9	illustration de la segmentation
Figure 10	architecture de YOLO
Figure 12	illustration de l'output de YOLO
Figure 13	architecture de SSD.
Figure 14	The architecture of R-CNN. (Image source : Girshick et al., 2014)
Figure 15	Illustration de Faster R-CNN model. (Image source : Ren et al., 2016)
Figure 16	Yolo dans colab
Figure 17	output Yolo
Figure 18	script ssd
Figure 19	output SSD
Figure 20	script YOLACT
Figure 21	output Yolact
Figure 22	YOLOv4 atteint l'état de l'art pour la détection d'objets
Figure 23	les photos de la base de données
Figure 24	La photo sans masque
Figure 25	La photo avec cadre de détection
Figure 26	les repères faciaux
Figure 27	les repères faciaux à l'aide de dlib afin que nous sachions où placer un masque sur le visage.
Figure 28	Un exemple de masque/écran facial COVID-19/Coronavirus.
Figure 29	photo de visage insérer avec masque de visage
Figure 30	photo de source de training
Figure 31	Les algorithmes de mask detection.
Figure 32	Output image mask detection
Figure 33	Output video mask detection
Figure 34	image d'une photo de caméra thermique

Figure 35	Classification des pixels en segments
Figure 36	Capteur thermique et la résultat d'utilisation
Figure 37	logo de projet
Figure 38	bête a cornes de projet
Figure 39	diagramme pieuvre de « Smart ascenseur »
Figure 40	Tableau des fonctions de service.
Figure 41	image de RaspBerry Pi 4
Figure 42	tableau fiche technique du caméra thermique testo

Liste des Abréviations :

CHU : centre hospitalier universitaire de Fès.

IA : l'intelligence artificielle.

CV : la vision par ordinateur.

ML : machine Learning.

DL : Deep Learning.

IBM : est une société multinationale américaine présente dans les domaines du matériel informatique.

YOLO : You Only Look Once.

YOLACT : You Only Look At Coefficients.

GPU : Graphics processing unit

TPU: Tensor Processing Unit. Custom build ASIC to accelerate TensorFlow projects.

SSD : (Single Shot MultiBox Detector)

Introduction général

Récemment le monde entier a été confronté à la pandémie " coronavirus " qui a engendré de nombreux dégâts humains et qui a atteint plus de 186 pays. Le Maroc, depuis le 15 mars 2020 a mis en place des mesures pour limiter la propagation de la pandémie.

Des études ont montré que plus de 80% des contaminations ne se font que par contact entre personnes, ou en touchant un objet contaminé par le virus, en revanche les mesures de protection peuvent prévenir 99% des cas c'est pour cela plus de la moitié de la population a été confinée. Dans ce travail on essayera d'expliquer comment on peut exploiter les nouvelles technologies, surtout l'intelligence artificielle, afin d'éradiquer ou du moins prévenir le virus covid-19.

Parmi les endroits où la prévention est de mise et où la vulnérabilité des personnes est très élevée, c'est les centres hospitaliers universitaires (CHU). En effet, les CHU connaissent déjà en période normale une grande affluence qui favorise une circulation du virus, et aussi les patients sont plus exposés au virus. Aussi, la circulation du virus est plus facile dans les endroits où la concentration des personnes est élevée. Les ascenseurs sont des endroits où le risque de transmission est très élevé, et nécessitent donc une prévention particulière.

Généralement les établissements de soins sont dotés d'ascenseurs fiables et confortables. Tous les flux logistiques doivent être en parfaite harmonie. Tous les visiteurs doivent être sous contrôle médicale....

Ce travail s'inscrit dans ce contexte, où il est proposé un ascenseur intelligent « Smart Ascenseur » capable de réagir par rapport aux situations favorisant contamination ou transmission.

La technologie Smart ascenseur est alimentée par une informatique très avancées, y compris l'intelligence artificielle (IA), la vision par ordinateur (CV) et le edge computing. Cette puissance de calcul est ce qui rend les technologies plus avancées que les autres systèmes de sécurité ; ils peuvent surveiller, identifier, alerter et parfois même prévenir les incidents critiques pour la sécurité en temps réel.

Ce rapport est structuré comme suit :

Un premier chapitre situant le contexte du travail, le deuxième traite la détection des personnes et la surveillance au CHU, le troisième la détection de la bavette et la mesure de la Température des personnes grâce à l'intelligence artificielle, et pour finir la discussion sur le projet de mon stage avec un don des solutions suggérées pour la résolution des problèmes rencontrés, le chapitre quatre va en faire l'affaire.

Chapitre 1 : présentation du contexte générale

1.Présentation générale de centre hospitalier universitaire de Fès (CHU)

Un centre hospitalier universitaire (CHU) est un hôpital lié à une université. Cet hôpital est soit un service de l'université, soit une entité distincte liée à cette dernière par une convention. Le CHU peut ainsi permettre la formation théorique et pratique des futurs professionnels médicaux, personnels paramédicaux et chercheurs en sciences de la santé.

Le Centre Hospitalier Universitaire Hassan II de Fès est un établissement public doté de la personnalité morale et de l'autonomie financière. Il est soumis à la tutelle du Ministère de la Santé. Il a été créé en vertu de la Loi 82.00 promulguée par le Dahir 1. Il a été inauguré le 14 janvier 2009 par le roi Mohammed VI, sur une superficie de 12 hectares, et est sous la juridiction du ministère marocain de la Santé. Il se compose de 5 hôpitaux: Hôpital des Spécialités, Hôpital Mère-Enfant, Hôpital d'Oncologie, Hôpital Omar Drissi, Hôpital Ibn Al Hassan. Il dispose en plus d'un espace au service des familles des malades des urgences, d'un centre d'addictologie et d'un centre au service de lutte contre la dépendance.

Cet édifice sanitaire, prévu pour répondre aux besoins de plus de quatre millions d'habitants (Régions Fès Boulemane, Meknès-Tafilalet et Taza-Al Hoceima-Taounate), a pour objectif d'améliorer le taux de couverture médicale de cette population et de décongestionner les structures sanitaires déjà existantes dans ces régions.

La dimension environnementale n'a pas été omise, puisque les responsables du CHU de Fès s'évertuent à certifier les prestations ayant un impact sur la sauvegarde de son environnement et à renforcer le traitement des déchets hospitaliers.

2.historique

Les travaux de construction du CHU Hassan II de Fès ont démarré fin novembre 2001 et c'est en janvier 2009 que le nouveau complexe hospitalier a été inauguré par SM le Roi Mohammed VI.[4]

Le plan d'action 2011 du CHU décline sa stratégie globale en six axes, à savoir l'amélioration de la qualité de la prise en charge des patients, le renforcement de l'offre tertiaire de soins et l'amélioration de la gouvernance, le renforcement de l'hygiène hospitalière, la modernisation des outils de gestion et le développement du professionnalisme, la mise en place d'une politique environnementale et de développement durable et, enfin, la promotion de la formation et de la recherche

Le Centre hospitalier universitaire (CHU) Hassan II de Fès a été choisi meilleur centre hospitalier maghrébin et 10ème au niveau africain par le site web spécialisé "Webometrics Hospitals", initié par le groupe de recherche espagnol indépendant "Cybermetrics Lab ».

Au niveau arabe, le CHU Hassan II de Fès a occupé la 19-ème place, derrière des hôpitaux saoudiens, dont l'hôpital et centre de recherche spécialisé Roi Fayçal, qui trône au classement arabe, et émiratis notamment, mais devançant avec des écarts fort significatifs plusieurs cliniques et hôpitaux d'Egypte, du Maroc et de l'Algérie.

3. Introduction de l'intelligence artificielle

3-1. L'intelligence artificielle (IA)

Nombreuses sont les définitions qu'on peut trouver de l'AI, un seul dictionnaire par exemple, on donne quatre définitions de l'intelligence artificielle :

-Un domaine d'étude dans le domaine de l'informatique. L'intelligence artificielle est concernée par le développement d'ordinateurs capables de s'engager dans des processus de pensée humains tels que l'apprentissage, le raisonnement et l'autocorrection.

Donc le but du projet Smart Ascenseur c'est arriver à des machines 'Intelligentes' qui prendront la décision au lieu de l'être humain (chef de service). Lorsqu'on parle de l'IA on ne peut pas passer sans aborder ses outils à savoir le Machine Learning ML et le Deep Learning DL.[1]

3-2 Machine Learning

Le machine Learning (ML), en français apprentissage automatique ou encore apprentissage statistique, est un sous-domaine de l'intelligence artificielle (IA) qui permet à des applications de prédire des résultats de plus en plus précis sans être explicitement programmées en ce sens.

L'apprentissage automatique se concentre sur les applications qui apprennent de l'expérience et améliorent leur prise de décision ou leur précision prédictive au fil du temps. [2]

Les étapes de formation d'un modèle avec le ML :

Étape 1 : Sélection et préparation d'un ensemble de données de formation

Étape 2 : Choix d'un algorithme à exécuter sur l'ensemble de données de formation

Étape 3 : Entraînement de l'algorithme pour créer le modèle

Étape 4 : Utilisation et amélioration du modèle

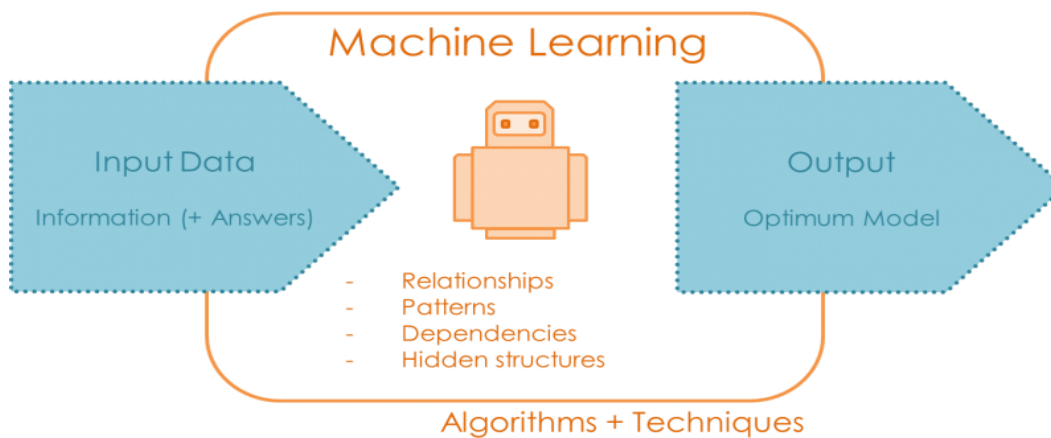


Figure 1: illustration du fonctionnement du ML

3-3 Méthode d'apprentissage automatique

Les méthodes d'apprentissage automatique (aussi appelées machine Learning styles) sont de trois catégories principales.

3-2-1.Apprentissage automatique supervisé : Supervised ML

-L'apprentissage automatique supervisé s'entraîne sur un ensemble de données étiquetées, c'est à dire, on a les inputs et leurs résultats. Autrement dit, les données sont étiquetées avec des informations que le modèle

d'apprentissage automatique est en cours de construction pour les déterminer et que le modèle est censé les classer. Par exemple, un modèle de vision par ordinateur conçu pour identifier les chiens et les chats pourrait être formé sur un ensemble de données de diverses images de chiens et chats étiquetés.

-L'apprentissage automatique supervisé nécessite moins de données de formation que les autres méthodes d'apprentissage automatique et facilite la formation, puisque les résultats du modèle peuvent être comparés aux résultats réels étiquetés

-L'apprentissage supervisé est généralement effectué dans le contexte de la classification et de la régression.

Régression

Un problème de régression se pose lorsque la variable de sortie est une valeur réelle, telle que « Dirham » ou « une distance ».

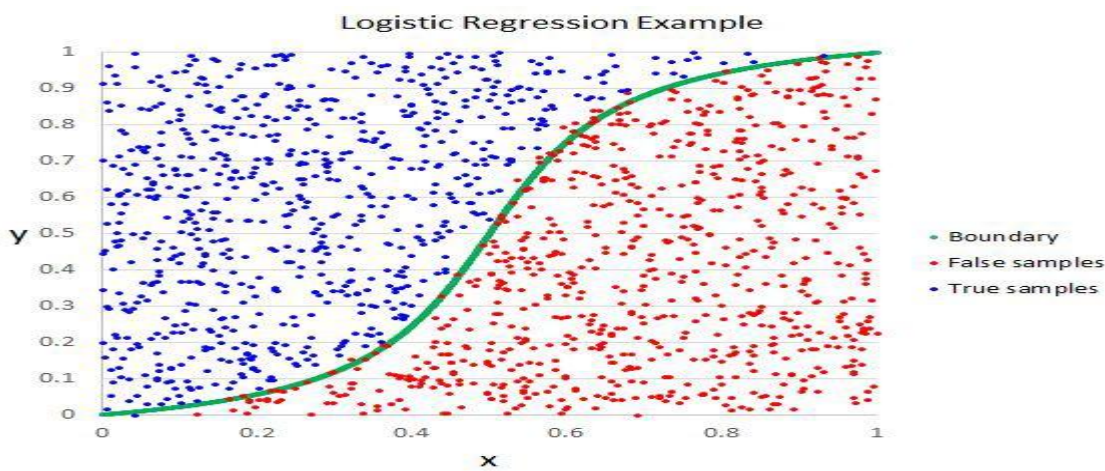


Figure 2 : un exemple d'un problème de régression

Classification

Un problème de classification survient lorsque la variable de sortie est une catégorie c'est à dire discrète, telle que «rouge», «bleu» ou «maladie» et «pas de maladie». :

Détection de courrier électronique indésirable (spam, pas spam).

En médecine, pour prédire si un patient a une maladie particulière ou non.

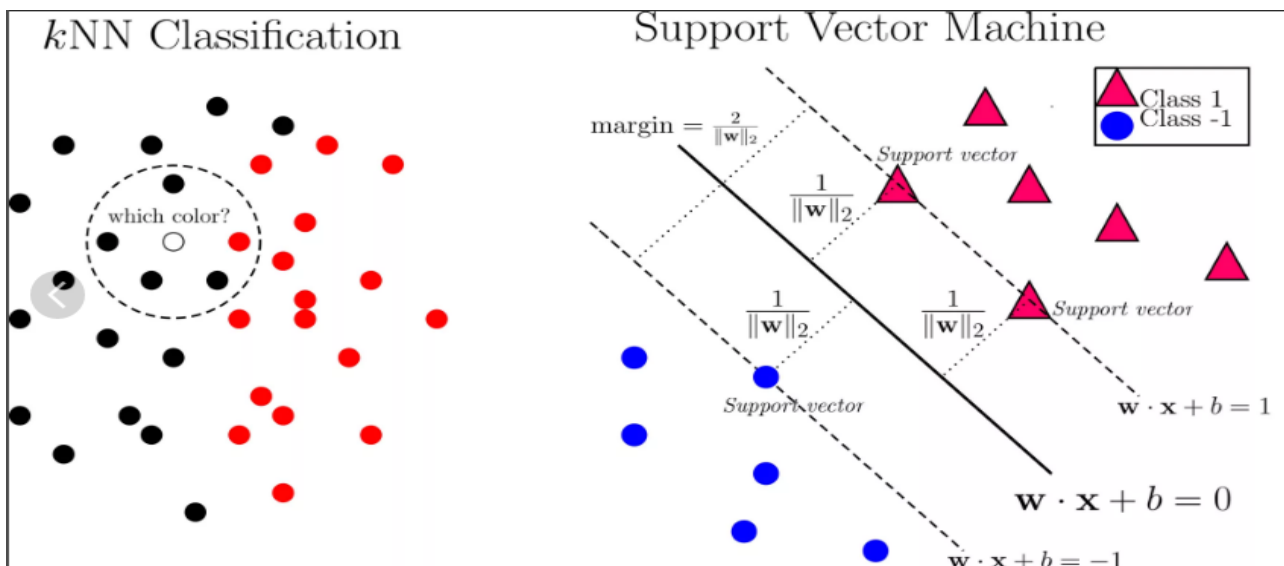


Figure 3 : exemple d'un problème de classification

3-3-2.Apprentissage automatique non supervisé : Unsupervised ML

L'apprentissage non supervisé ingère des données non étiquetées dont on ne connaît pas la réponse correcte et utilise des algorithmes pour extraire les fonctionnalités significatives nécessaires pour étiqueter, trier et classifier les données en temps réel, sans intervention humaine. L'apprentissage non supervisé est moins une question d'automatisation des décisions et des prédictions et plutôt d'identification des modèles et des relations dans les données que les êtres humains manqueraient. Prenons la détection des spams, par exemple les gens génèrent plus d'e-mails qu'une équipe de scientifiques de données ne pourrait jamais espérer étiqueter ou classer au cours de leur vie. Un algorithme d'apprentissage non supervisé peut analyser d'énormes volumes d'e-mails et découvrir les fonctionnalités et les modèles qui indiquent le spam (et continuer à s'améliorer à signaler le spam au fil du temps).

3-3-3.Apprentissage automatique de renforcement : Reinforcement ML

L'apprentissage automatique de renforcement est un modèle d'apprentissage automatique

Comportemental qui est similaire à l'apprentissage supervisé, mais l'algorithme n'est pas formé à l'aide de données d'échantillon. Ce modèle apprend au fur et à mesure qu'il utilise les tentatives et erreurs. Une séquence de résultats positifs seront renforcés afin d'élaborer la meilleure recommandation ou politique pour un problème donné. Le système IBM Watson® qui a remporté le *Jeopardy* ! défi en 2011 en est un bon exemple. Le système a utilisé l'apprentissage de renforcement pour décider s'il faut tenter une réponse et quel carré choisir sur le tableau.

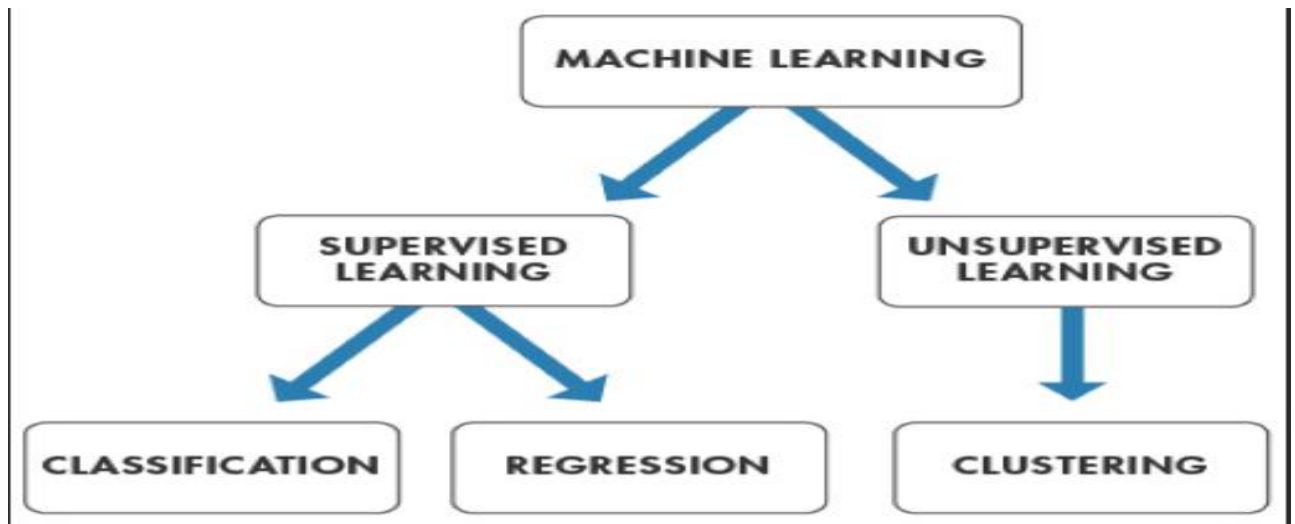


Figure4 : les types du ML.

3-4. Deep Learning : DL

L'apprentissage profond est une forme d'intelligence artificielle, dérivée du Machine Learning, c'est une fonction d'IA qui imite le fonctionnement du cerveau humain dans le traitement des données à utiliser pour détecter des objets, reconnaître la parole, traduire des langues et prendre des décisions. Le DL est capable d'apprendre sans supervision humaine, en s'appuyant sur des données à la fois non structurées et non étiquetées

Intelligence artificielle aide le projet « Smart ascenseur » à scanner et surveiller le passage pour informer l'unité de traitement avec des informations instantanées. Smart ascenseur utilise des algorithmes d'apprentissage profond pour améliorer la clarté d'image des visiteurs, puis automatisée des événements de classification. Dans notre projet on va traiter la partie qui concerne la détection et la classification des utilisateurs.

Chapitre 2 : la détection des personnes et la surveillance dans CHU

1 Le principe de fonctionnement

La détection d'objets est un domaine très actif de la recherche qui cherche à classer et localiser des régions/zones d'une image ou d'un flux vidéo. Ce domaine est à la croisée de deux autres : la classification d'image et la localisation d'objets. En effet, le principe de la détection d'objets est le suivant : pour une image donnée, on recherche les régions de celle-ci qui pourraient contenir un objet puis pour chacune de ces régions découvertes, on l'extrait et on la classe à l'aide d'un modèle de classification d'image. Les régions de l'image d'origine ayant de bons résultats de classification sont conservées et les autres jetées. Ainsi, pour avoir une bonne méthode de détection d'objets, il est nécessaire

d'avoir un algorithme solide de détection de régions et un bon algorithme de classification d'images. Le deep Learning (et notamment les réseaux de convolution) est devenue la méthode numéro #1 pour résoudre ce genre de problème. (On consacre toute une partie après pour expliquer comment ça marche). Les recherches en détection d'objets ont donc tout naturellement intégré les modèles de classification d'image, ce qui a permis de créer des très bons algorithmes tels que : YOLO, R-CNN, SSD... Ces algorithmes sont basés principalement sur le Deep Learning ou bien ce qu'on appelle les réseaux de neurones.

Exemple :

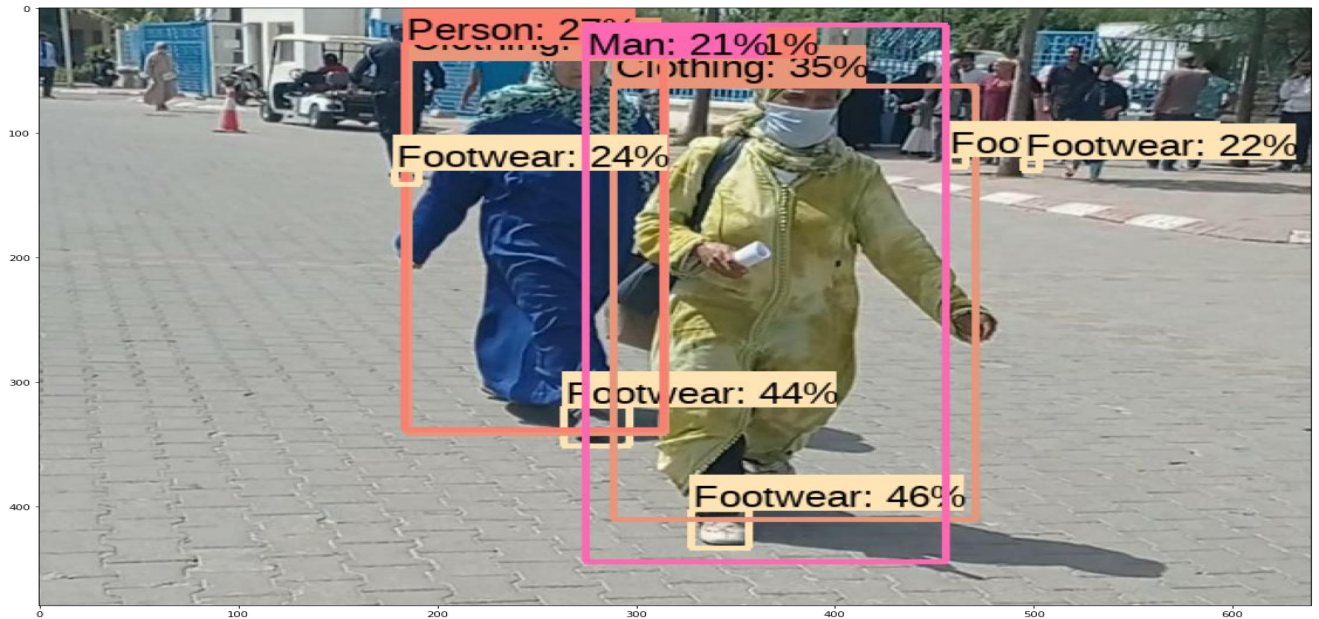


Figure 5: image qui illustre le résultat d'une détection d'objets

La question, qui se pose est : en quoi consiste la détection des objets en utilisant le deep Learning ?

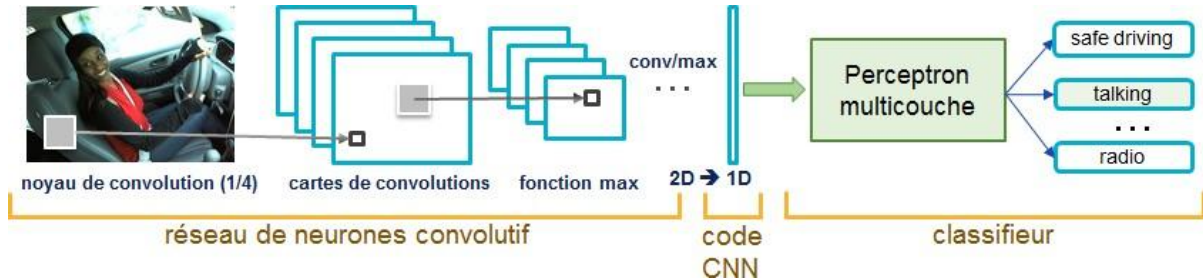
1-1. Le rôle du Deep Learning dans la détection des objets

Le concept de Deep Learning (et plus généralement Machine Learning), dit en français "apprentissage profond", remonte au milieu du 20ème siècle. A l'époque, des scientifiques imaginent déjà des machines "capable d'apprendre". Pour ce faire, différentes techniques ont été pensées et développées pour générer des algorithmes en mesure d'apprendre mais aussi de s'améliorer de manière autonome. On retrouve principalement comme technique les réseaux de neurones artificiels qui s'inspirent clairement des neurones du cerveau humain.

Les réseaux de neurones convolutifs sont à ce jour les modèles les plus performants pour classer des images. Désignés par l'acronyme CNN, de l'anglais Convolutional Neural Network, ils comportent deux parties bien distinctes. En entrée, une image est fournie sous la forme d'une matrice de pixels. Elle a 2 dimensions pour une image en niveaux de gris. La couleur est représentée par une troisième dimension, de profondeur 3 pour représenter les couleurs fondamentales [Rouge, Vert, Bleu].

La première partie d'un CNN est la partie convolutive à proprement parler. Elle fonctionne comme un extracteur de

caractéristiques des images. Une image est passée à travers une succession de filtres, ou noyaux de convolution, créant de nouvelles images appelées cartes de convolutions. Certains filtres intermédiaires réduisent la résolution de l'image par une opération de maximum local. Au final, les cartes de convolutions sont mises à plat et concaténées en un vecteur de caractéristiques, appelé code CNN.



Ce code CNN en sortie de la partie convolutive est ensuite branché en entrée d'une deuxième partie, constituée de couches entièrement connectées (perceptron multicouche). Le rôle de cette partie est de combiner les caractéristiques du code CNN pour classer l'image.

La sortie est une dernière couche comportant un neurone par catégorie. Les valeurs numériques obtenues sont généralement normalisées entre 0 et 1, de somme 1, pour produire une distribution de probabilité sur les catégories.

Pour plus de détails et d'explications de ces réseaux de neurones et le jargon associé, on propose la partie suivante :

Le principe de réseau de neurones

On a démontré l'utilité pertinente des réseaux de neurones dans la conception des algorithmes de détection des objets, mais le problème qui a poussé les chercheurs à inventer ce réseau c'était comment on peut imiter l'intelligence humaine et être flexible dans la prise de décision, même si toutes les situations possibles ne sont pas précisées pour l'algorithme. Pour bien illustrer ça, on prend l'exemple des images à droite, le cerveau humain n'a aucun mal à reconnaître que c'est trois, dans les trois images, tous sont reconnaissable à des trois, même si les valeurs spécifiques de chaque pixel sont très différentes d'une image à l'autre. Le challenge est d'essayer de simuler cette procédure de détection des éléments qui sont différents mais représentent la même chose.

La solution établie par le Deep Learning, est ce qui est appelé par le réseau de neurones, qui est un système dont la conception est à l'origine schématiquement

Inspirée du fonctionnement des neurones biologiques. Ce réseau est constitué d'un ensemble de neurones interconnectés permettant la résolution du problème comme dans la figure, on trouve que 2 est la valeur la plus probable de l'image à l'entrée du réseau, ce qui est vrai.

En règle générale, un réseau de neurones repose sur un grand nombre de processeurs opérant en parallèle et organisés en tiers. Le premier tiers reçoit les entrées d'informations brutes, un peu comme les nerfs optiques de l'être humain lorsqu'il traite des signaux visuels.

Dans notre exemple, la dernière couche contient 10 neurones, chacun représente un chiffre, et la coloration de



chaque neurone dépend d'un nombre appelé l'activation du neurone, qui est un nombre entre 0 et 1, et qui représente à quel point le système pense que l'image correspond à un chiffre donné. Il y a aussi quelques couches entre, appelées les "couches cachées", qui contient aussi des neurones, chacune est associée à son activation.

Pour la première couche, le nombre des neurones est le nombre des pixels de l'image à l'entrée, selon la luminosité de chaque pixel dans l'image, ce motif d'activations entraîne un autre motif spécifique dans la couche suivante, qui entraîne un autre motif dans celle juste après, qui au final donne un dernier motif dans la dernière couche.

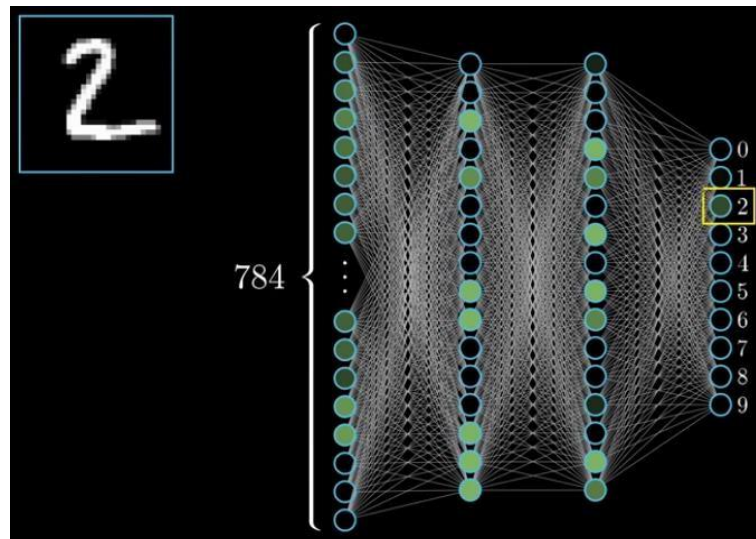


Figure 6: réseau de neurones

Le cœur du réseau en tant que mécanisme de gestion d'informations revient à "Comment l'activation d'une couche entraîne l'activation de la couche suivante ?"

Pourquoi une structure comme celle-ci peut agir intelligemment ?

Toute entité complexe peut être décomposée en éléments simples, par exemple, et restant dans notre exemple, le chiffre 9 a une boucle en haut, et une ligne à droite, et les autres chiffres aussi sont décomposés de la même manière :

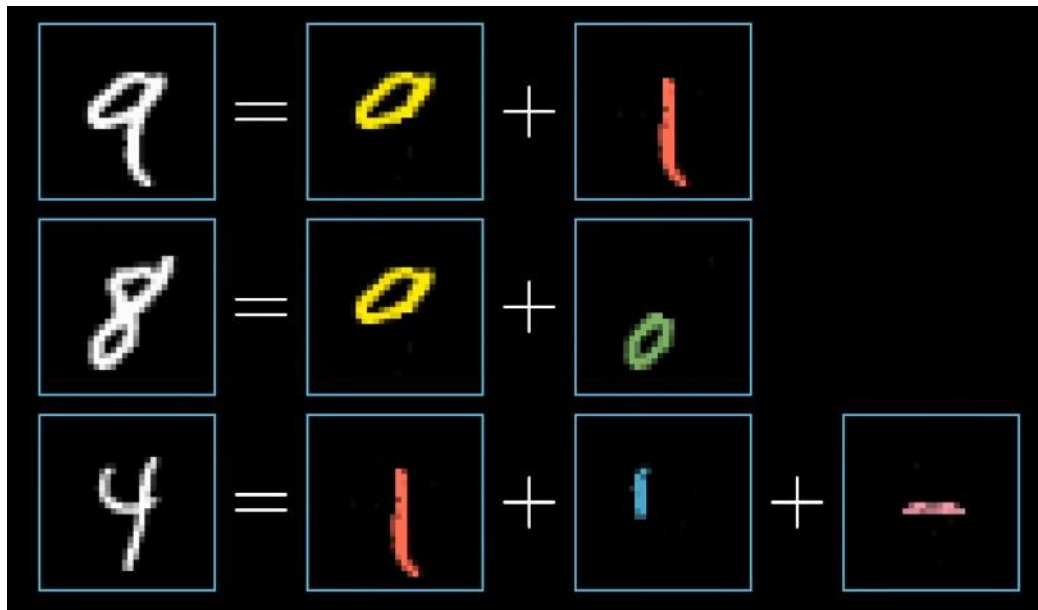
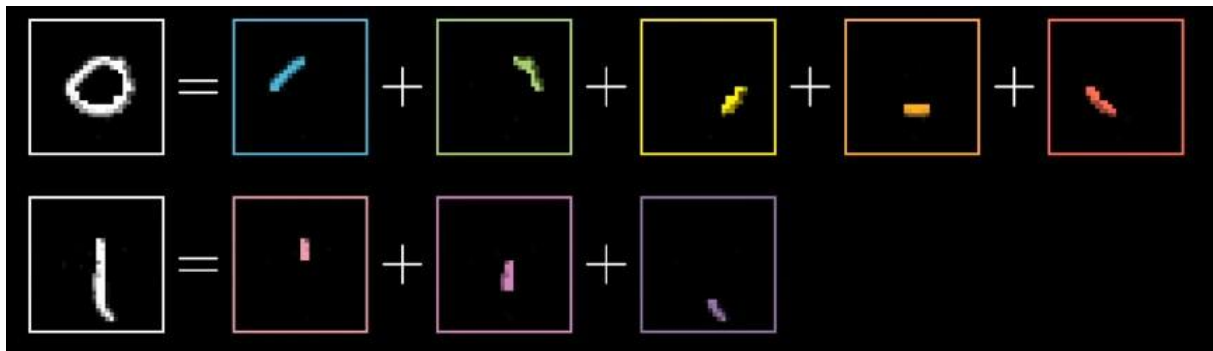


Figure 7: séparation de l'input par le réseau

Donc, ces formes simples doivent être modélisé dans la troisième couche, et chaque fois on donne à ce réseau une image qui contient une boucle, par exemple 8 ou 9, il y a certains neurones spécifiques dont l'activation sera proche de 1. Donc le passage de la troisième couche à la dernière ne requit qu'un apprentissage de : Quelle combinaison de formes correspond à quel chiffre ?

Ces formes simples peuvent aussi être décomposées en des formes plus simples,



Et suivant cette approche, on continuera de construire les autres couches.

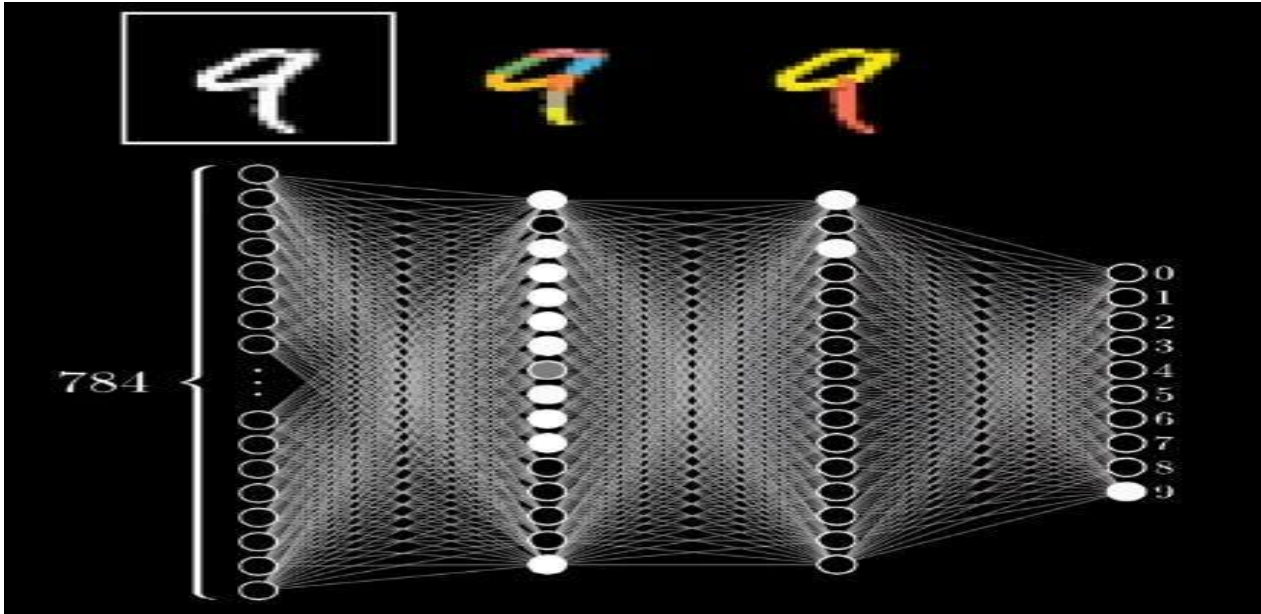


Figure 8: CNN layers

Maintenant, mathématiquement parlant, comment on peut détecter l'existence de ces formes dans l'image ? Et quels paramètres devraient avoir le réseau pour que ce soit capable de capturer le motif ?

Nous allons donc assigner un poids à chacune des connections entre nos neurones et les neurones de la première couche, ces poids sont juste des nombres, et on prend la somme pondérée des activations de la première couche en accord avec ces poids pour l'affecter à l'activation de la couche suivante.

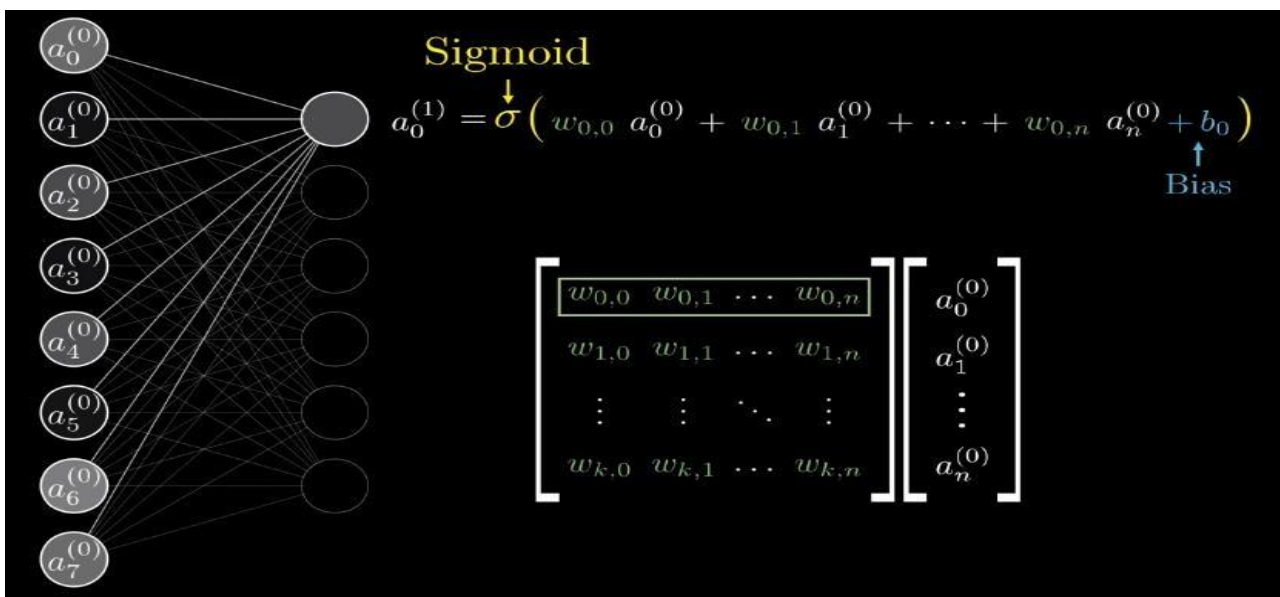


Figure 9: illustration de la segmentation

2.Exemples des algorithmes de détection et de classification utilisés

2-1.Génération du Bounding Box

Le model basé sur les CNN du DL ensuite est entraîné plusieurs fois pour pouvoir détecter les objets dans les images. Dans un premier temps il fait une classification.



Classificatrice multi-étiquette

La classification d'image prend une image et prédit l'objet dans cette image. Par exemple, lorsqu'on construit un classificateur on prend des images et on prédit leur classe :

Pour résoudre ce problème, il y a formation d'une classificatrice multi-étiquette qui prédira à la fois les classes. Ensuite il va détecter l'emplacement de l'objet dans l'image. Le problème de l'identification de l'emplacement d'un objet (compte tenu de la classe) dans une image est appelé localisation.

Prédire l'emplacement de l'objet avec la classe est appelé détection d'objet. Maintenant il y'a prédiction des classes ainsi des rectangles (appelés Bounding Boxes) contenant les objets. Il faut 4 variables pour identifier un rectangle de façon unique. Ainsi, pour chaque instance de l'objet dans l'image, nous prévoyons les variables suivantes :

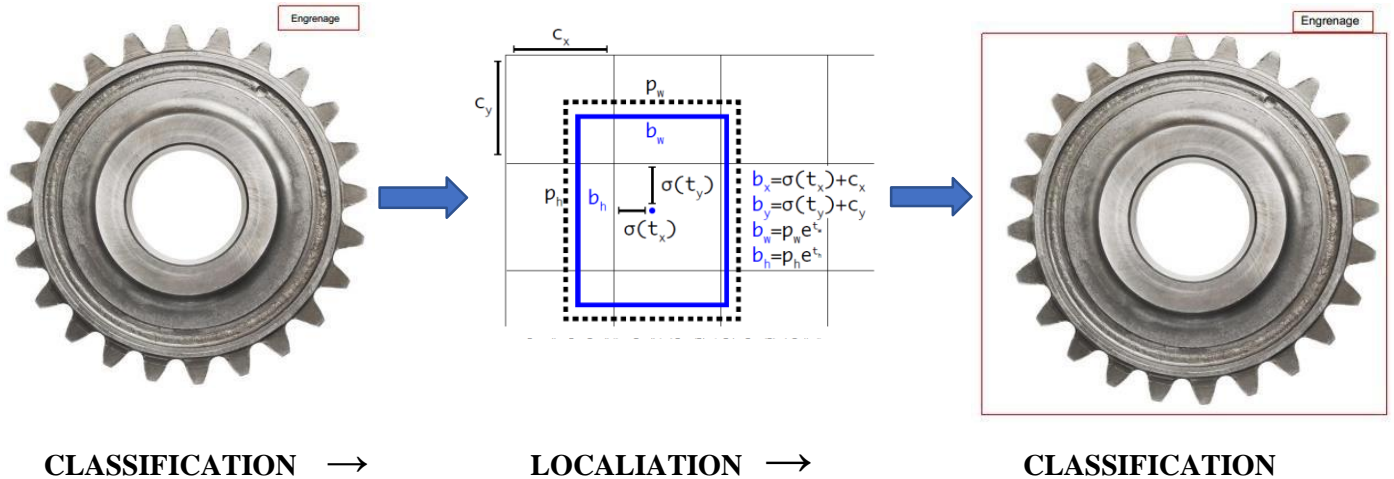
Class_name.

Bounding_box_top_left_x_coordinate.

Bounding_box_top_left_y_coordinate.

Bounding_box_width.

Bounding_box_height.



Il existe différentes méthodes de détection d'objets comme RCNN, Faster-RCNN, SSD etc. Pourquoi avons-nous tant de méthodes et quelles sont les principales caractéristiques de chacune d'entre elles ?

2-2.YOLO (YOU ONLY LOOK ONCE : Vous ne regardez qu'une seule fois)

YOLO est venu sur la scène de la vision par ordinateur CV avec l'article fondateur de 2015 par Joseph Redmon. « You Only Look Once : Unified, Real-Time Object Détection, » et a immédiatement obtenu beaucoup d'attention par d'autres chercheurs en vision informatique.

YOLO suggère d'avoir un réseau unifié pour effectuer tout comme une fois. En outre, le réseau de formation de bout en bout peut être atteint.

LA Détection de l'image :

L'image d'entrée est divisée en une $S \times S$ ($S=7$). Si le centre d'un objet tombe dans une cellule de grille, cette cellule réseau est responsable de la détection de cet objet. Chaque cellule de grille prédit les Bounding Boxes B ($B=2$) et les scores de confiance pour ces boîtes. Ces scores de confiance reflètent la confiance du modèle que la boîte contient un objet, c'est-à-dire tous les objets dans la boîte, P (Objets)

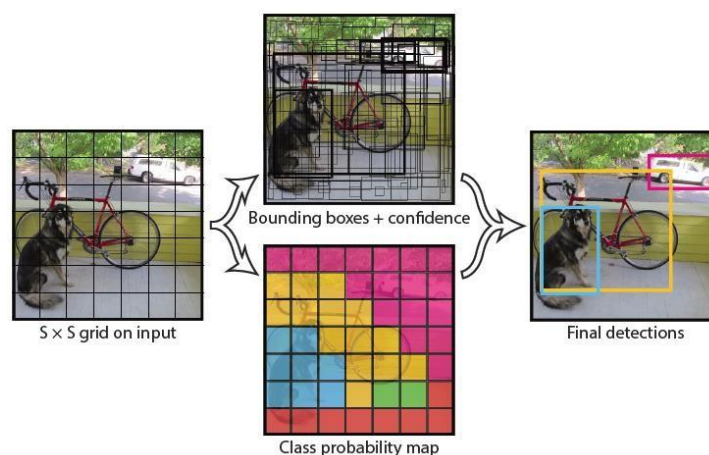


Figure 10: architecture de YOLO

L'output de YOLO peut être illustré comme suit :

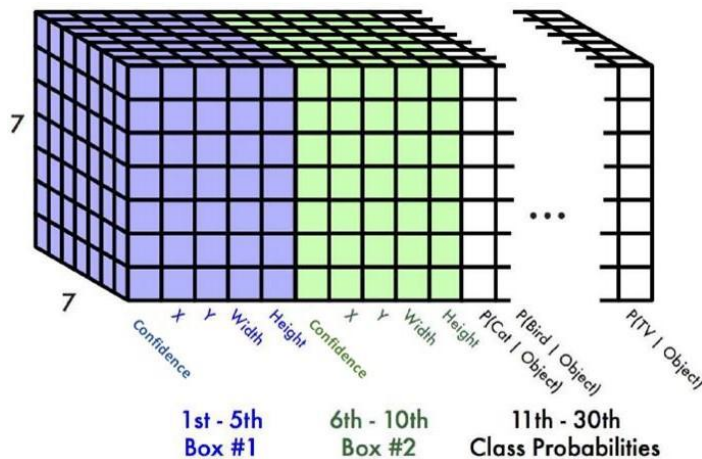


Figure 11: illustration de l'output de YOLO

2-3.SSD (Single Shot MultiBox Detector) :

Single Shot MultiBox Detector (SSD) est sorti quelques mois après YOLO comme une alternative digne. De même que YOLO, la détection d'objets se fait en une seule propagation vers l'avant du réseau. Ce modèle CNN de bout en bout transmet l'image d'entrée à travers une série de couches convolutionnelles, générant des boîtes de déliement des candidats à différentes échelles en cours de route. Comme vérité de base pour la formation, SSD considère les objets étiquetés comme des exemples positifs, et toutes les autres boîtes bondissantes qui ne se chevauchent pas avec les points positifs sont des exemples négatifs. Il s'avère que la construction de l'ensemble de données de cette façon le rend très déséquilibré.

Pour cette raison, SSD applique une méthode appelée exploitation minière négative dure (hard négative

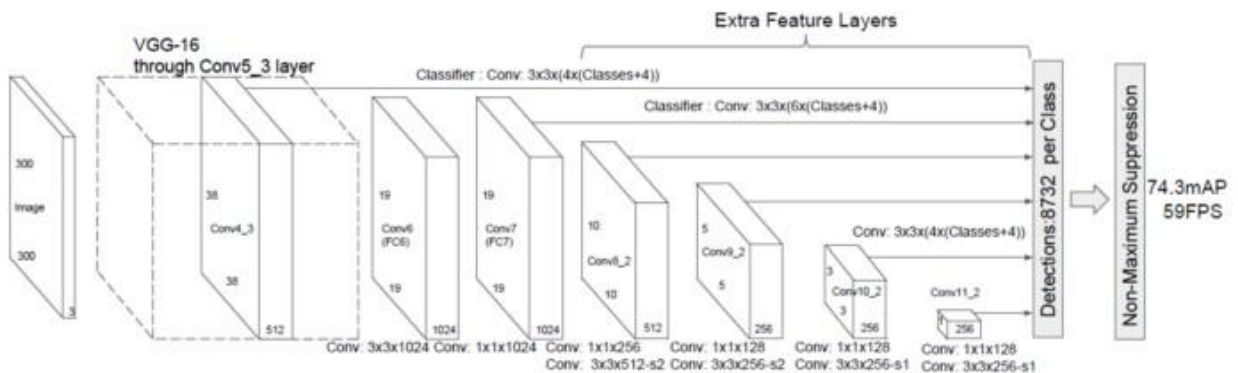


Figure 12: architecture de SSD.

mining) juste après avoir effectué NMS (non-max suppression). L'exploitation minière négative dure est une méthode pour choisir seulement les exemples négatifs avec la perte de confiance la plus élevée, de sorte que le rapport entre les positifs et les négatifs est tout au plus 1 :3. Cela conduit à une optimisation plus rapide et à une phase de formation plus stable.

Dans l'image ci-dessus extrait du document officiel, nous pouvons voir que le réseau de base est VGG-16. Cependant, de nos jours, SSD est souvent utilisée avec un ResNet, Inception et même MobileNet qu'on a utilisé pour tester l'algorithme

2.4.R-CNN

R-CNN (Girshick et AL., 2014) est l'abréviation de « **Region-based Convolutional Neural Networks** ». Contourner le problème de la sélection d'un grand nombre de régions RCNN propose une solution composée de deux étapes. Tout d'abord, à l'aide d'une recherche sélective, il identifie un nombre gérable de candidats de la région objet bounding-box (« **Region of Interest**» ou « **RoI** »). Et puis il extrait les fonctionnalités de CNN de chaque région indépendamment pour la classification.

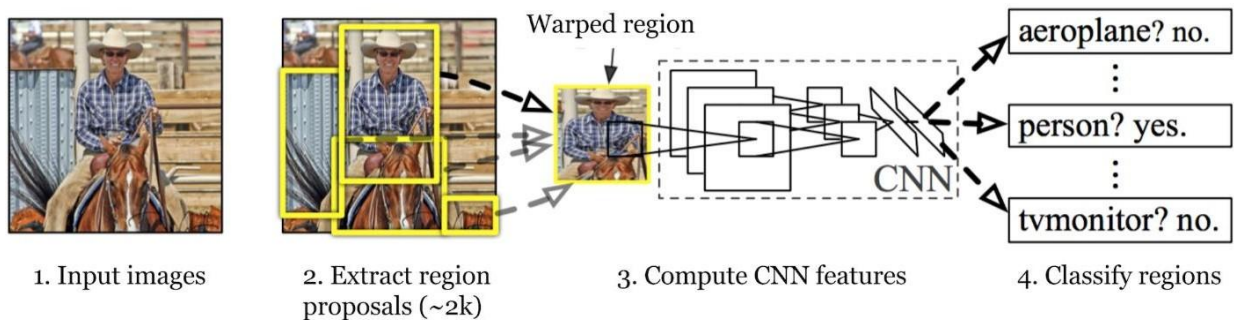


Figure 13: The architecture of R-CNN. (Image source : Girshick et al., 2014)

Faster R-CNN :

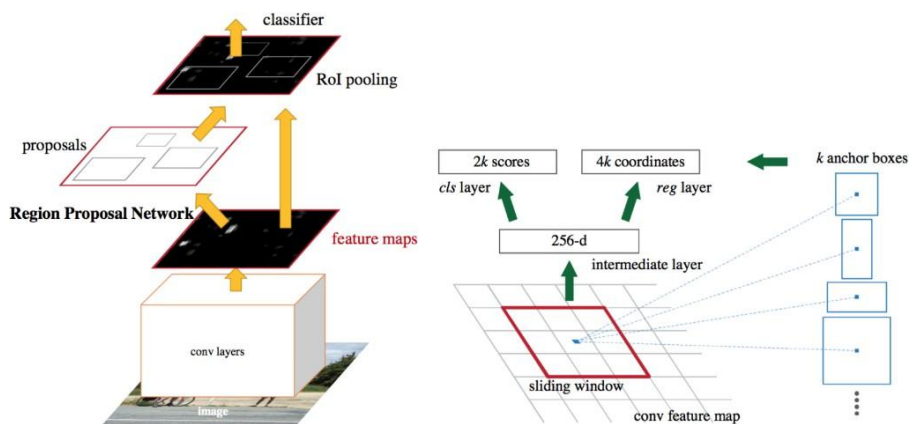


Figure 14: illustration de Faster R-CNN model. (Image source : Ren et al., 2016)

Une solution intuitive d'accélération consiste à intégrer l'algorithme de proposition de région dans le modèle CNN. C'est exactement ce que fait R-CNN plus rapide (Ren et al., 2016) : construire un modèle unique et unifié composé de RPN (réseau de proposition de région) et de R-CNN rapide avec des couches de caractéristiques convolutives partagées.

2.5. Yolact :

Pour comprendre son principe il faut d'abord comprendre les algorithmes qui génère des masks, Mask

Algorithmes prenons l'exemple de Mask Rcn, pour générer un mask qui délimite l'objet ont procédé par 2 étapes

La première étape (proposition de région) de Mask R-CNN est identique à son prédécesseur (Rcnn), tandis que dans la deuxième étape, il produit un masque binaire pour chaque RoI en parallèle à la classe et le Bounding box. Ce masque binaire indique si le pixel fait partie d'un objet, sans se préoccuper des catégories. La classe pour les pixels serait assignée simplement par la boîte de déliement dans laquelle ils résident, ce qui rend le modèle beaucoup plus facile à former.

Cependant, on rencontre quelques difficultés pour générer les masks :

- Segmentation est difficile, beaucoup plus difficile que la détection d'objets
- Elles nécessitent des quantités importantes de post-traitement après la localisation, et sont donc

Encore loin d'être en temps réel

Pour résoudre ces problèmes, nous proposons YOLACT (You Only Look At Coefficients), un cadre de segmentation d'instance en temps réel qui renonce à une étape de localisation explicite. Au lieu de cela, YOLACT décompose la segmentation des instances en deux tâches parallèles : (1) générer un dictionnaire de masques prototypes non locaux sur l'ensemble de l'image, et (2) prédire un ensemble de coefficients de combinaison linéaire par instance.

3.La surveillance dans CHU avec IA ((Yolo v5 , SSD avec mobile net ,Yolcat)

Google collaboration :

Colaboratory, souvent raccourci en "Colab », nous permet d'écrire et d'exécuter du code Python dans un navigateur qui a des avantages suivants :

- Aucune configuration requise
- Accès gratuit aux GPU(memoie RAM très fort , carte graphique puissante , stockage important..)
- Partage facile

Que nous soyons étudiant, data scientifique ou chercheur en IA, Colab peut nous simplifier la tâche. Avec l'option :

TPU : unité de traitement tensoriel. Créez un ASIC personnalisé pour accélérer les projets TensorFlow .

✓ Yolo v5:

L'implémentation a été faite sur Google Collaboration qui offre un GPU gratuit et très puissant (NVIDIA SMI : TESLA T4), après compilation on peut télécharger l'image ou la vidéo de l'output. [6]


```

YOLOv5.ipynb
File Edit View Insert Runtime Tools Help Changes will not be saved
+ Code + Text Copy to Drive
from google.colab import drive
drive.mount('/content/gdrive')
Drive already mounted at /content/gdrive; to attempt to forcibly remount, call drive.mount("/content/gdrive", force_remount=True).
Invidia-smi
Sat Mar 20 17:59:55 2021
+-----+
| NVIDIA-SMI 460.56      Driver Version: 460.32.03   CUDA Version: 11.2     |
+-----+-----+
| GPU Name      Persistence-M| Bus-Id        Disp.A   Volatile Uncorr. ECC | | |
| Fan  Temp  Perf  Pwr:Usage/Cap|      Memory-Usage  GPU-Util  Compute M. |
|              |                |    |               |
+-----+-----+
| 0   Tesla T4      Off          | 00000000:00:04:0 Off   |    0%      Default |
| N/A   36C    P8     9W /  70W   |  8MiB / 15109MiB |           | N/A   |
+-----+-----+
Processes:
GPU  GI  CI      PID  Type  Process name      GPU Memory
  ID  ID  ID                               Usage
+-----+-----+
No running processes found
+-----+-----+
[ ] !git clone https://github.com/ultralytics/yolov5
  
```

Figure 15 Yolo dans colab

On a testé sur des images qu'on a prises l'output sont les suivant dans la fst :

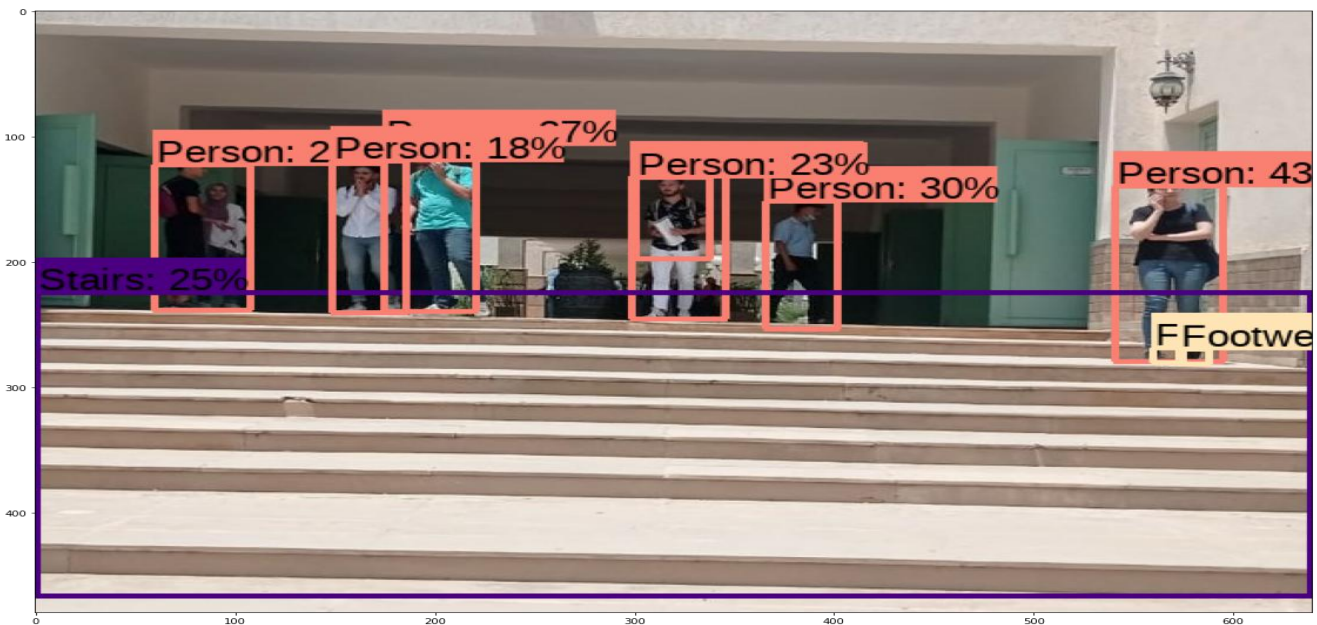


Figure16 : output Yolo

SSD avec mobile net :

Ces algorithmes ne nécessitent pas beaucoup de performance de la machine donc on l'a essayé sur notre PC (HP ; processeur i7 8th gen ; 8gb RAM)

```

jupyter mobilenet_ssd_python.py 12 hours ago
File Edit View Language

1 #Import the necessary libraries
2 import numpy as np
3 import argparse
4 import cv2
5
6 # construct the argument parse
7 parser = argparse.ArgumentParser(
8     description='Script to run MobileNet-SSD object detection network')
9 parser.add_argument("--video", help="path to video file. If empty, camera's stream will be used")
10 parser.add_argument("--prototxt", default='MobileNetSSD_deploy.prototxt',
11     help="Path to text network file: '
12     'MobileNetSSD_deploy.prototxt for Caffe model or '
13     ')
14 parser.add_argument("--weights", default='MobileNetSSD_deploy.caffemodel',
15     help="Path to weights: '
16     'MobileNetSSD_deploy.caffemodel for Caffe model or '
17     ')
18 parser.add_argument("--thr", default=0.2, type=float, help="confidence threshold to filter out weak detections")
19 args = parser.parse_args()
20
21 # Labels of Network
22 classNames = { 0: 'background',
23     1: 'aeroplane', 2: 'bicycle', 3: 'bird', 4: 'boat',
24     5: 'bottle', 6: 'bus', 7: 'car', 8: 'cat', 9: 'chair',
25     10: 'cow', 11: 'diningtable', 12: 'dog', 13: 'horse',
26     14: 'motorbike', 15: 'person', 16: 'pottedplant',
27     17: 'sheep', 18: 'sofa', 19: 'train', 20: 'tvmonitor' }
28
29 # Open video file or capture device.
30 if args.video:
31     cap = cv2.VideoCapture("video1.mp4")
32 else:
33     cap = cv2.VideoCapture("video1.mp4")
34
35 #Load the Caffe model
36
37 yLeftBottom = int(detections[0, 0, 1, 4] * rows)
38 yRightTop = int(detections[0, 0, 1, 5] * cols)
39 yRightTop = int(detections[0, 0, 1, 6] * rows)
40
41 # Factor for scale to original size of frame
42 heightFactor = frame.shape[0]/300.0
43 widthFactor = frame.shape[1]/300.0
44 # Scale object detection to frame
45 xLeftBottom = int(widthFactor * xLeftBottom)
46 yLeftBottom = int(heightFactor * yLeftBottom)
47 xRightTop = int(widthFactor * xRightTop)
48 yRightTop = int(heightFactor * yRightTop)
49 # Draw Location of object
50 cv2.rectangle(frame, (xLeftBottom, yLeftBottom), (xRightTop, yRightTop),
51     (0, 255, 0))
52
53 # Draw Label and confidence of prediction in frame resized
54 if class_id in classNames:
55     label = classNames[class_id] + ": " + str(confidence)
56     labelSize, baseline = cv2.getTextSize(label, cv2.FONT_HERSHEY_SIMPLEX, 0.5, 1)
57
58     yLeftBottom = max(yLeftBottom, labelSize[1])
59     cv2.rectangle(frame, (xLeftBottom, yLeftBottom - labelSize[1]),
60         (xLeftBottom + labelSize[0], yLeftBottom + baseline),
61         (255, 255, 255), cv2.FILLED)
62     cv2.putText(frame, label, (xLeftBottom, yLeftBottom),
63         cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 0, 0))
64
65     print(label) #print class and confidence
66
67 cv2.namedWindow("frame", cv2.WINDOW_NORMAL)
68 cv2.imshow("frame", frame)
69 if cv2.waitKey(1) >= 0: # Break with ESC
70     break

```

Figure 17 script ssd

L'output est le suivant en CHU:

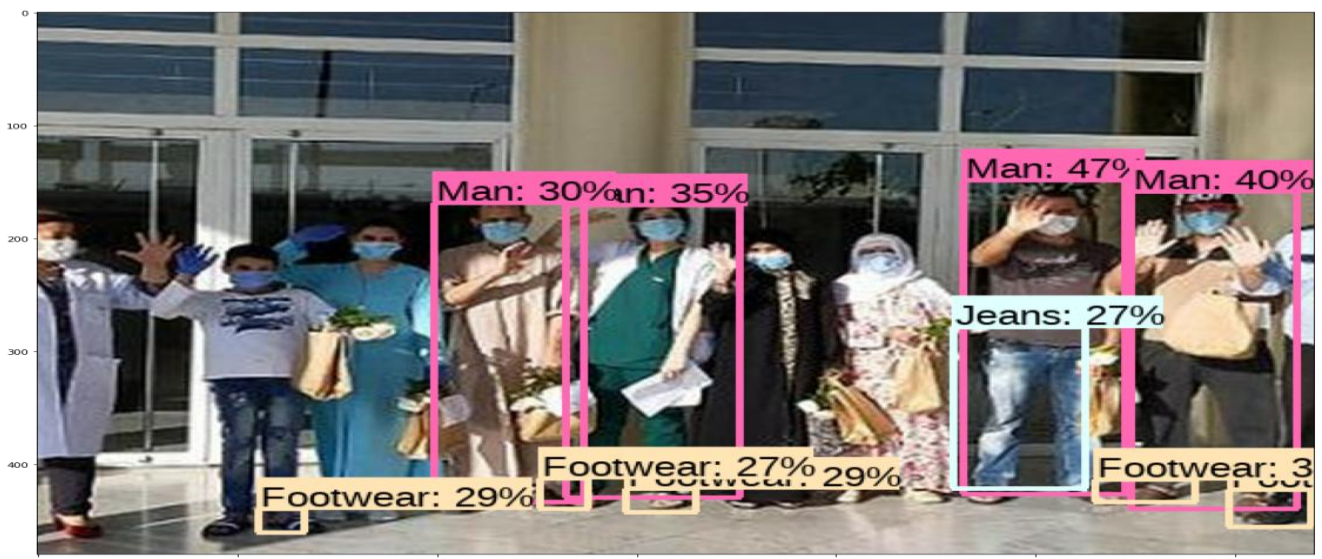


Figure 18 :output SSD

Yolcat : Comme on a mentionné avant on a utilisé ces algorithmes comme alternative de Mask-RCNN

```

YOLACT: Evaluating...
File Edit View Language

1 # Import the necessary libraries
2 import numpy as np
3 import cv2
4 import matplotlib.pyplot as plt
5 from pathlib import Path
6
7 output_images = Path('output_images')
8
9 def show_image(img_path):
10     img = cv2.imread(img_path)
11     img_cvt=cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
12     plt.figure(figsize=(16,16))
13     plt.imshow(img_cvt)
14     plt.show()
15
16 # Iterate through all of the output images and display them
17 for img_path in output_images.iterdir():
18     print(img_path)
19     show_image(str(img_path))
20
21 output_images/Crossing-street-pedestrians-road.png

```

Figure 19: script YOLACT

L'output est le suivant :

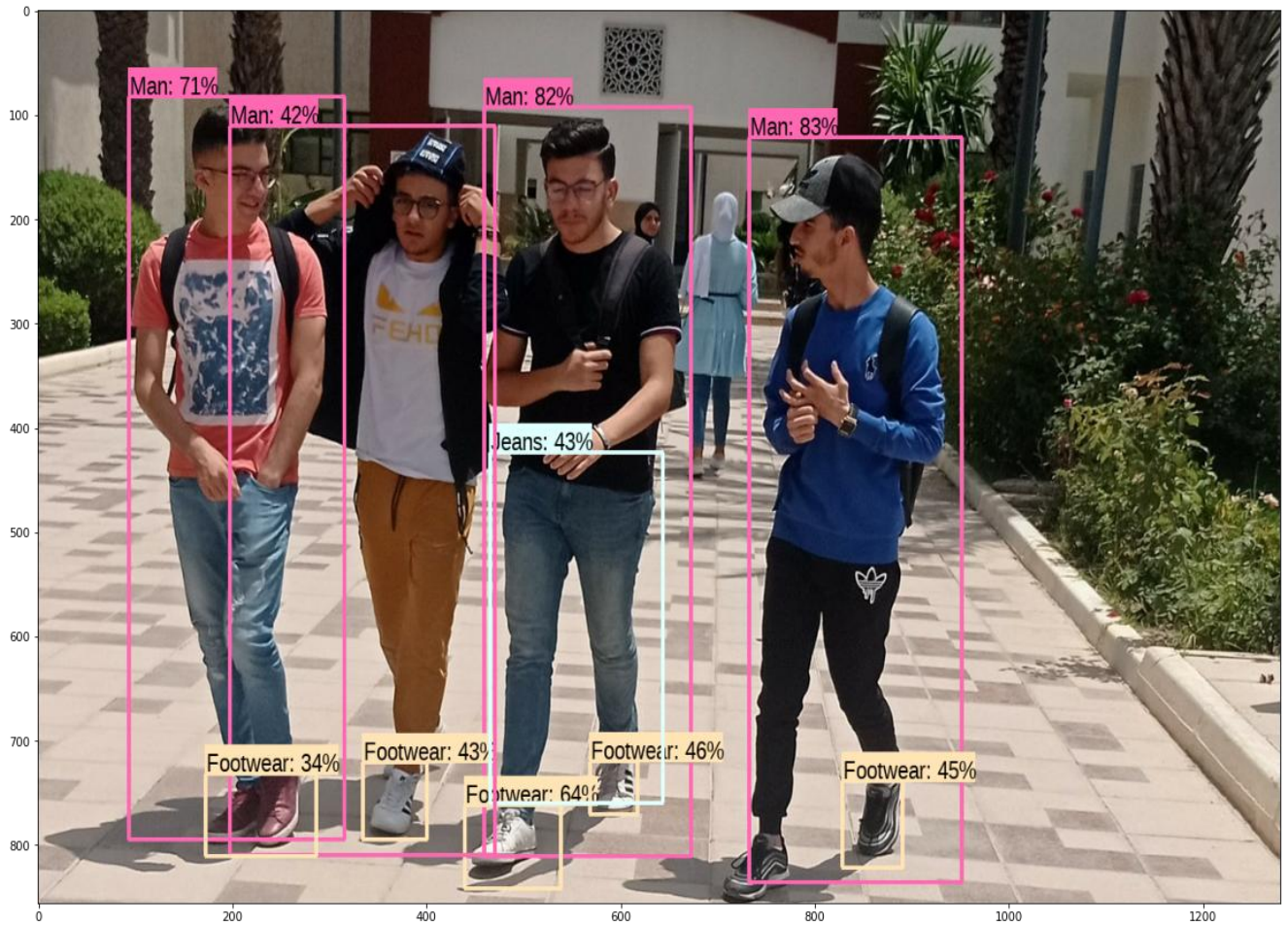


Figure 20: output Yolact in fst

La question, donc, qui se pose est : en quoi consiste la détection de covid en utilisant l'intelligence artificiel ? ET pourtant le travail que j'ai fait dans CHU, un modèle de deep Learning semble capable de réaliser un diagnostic différentiel de COVID-19 sur les smart ascenseur. Ce travail montre surtout que la spécificité de l'intelligence artificielle semble améliorée comparativement aux autres études publiées sur ce thème.

Chapitre 3 : la détection de la bavette et mesure de la Température des gens avec IA.

Depuis l'annonce de confinement, la reprise tu travail pour tous les secteurs d'activité va passer par la mise en place des gestes barrières pour palier à la propagation du COVID-19. Encadrés par un plan national de confinement, les gestes barrières, leurs coûts et leur mise en application peuvent devenir un vrai casse-tête à mettre en place.

Dans ce travail on s'est penché sur une solution de gestion d'accès (ou de file) permettant la détection du port d'un masque et prise de température sans contact. Asservi (ou pas) à une porte ou un tourniquet (Tripode) le dispositif peut gérer des flux de visiteurs, malades, employés efficacement et de manière autonome.

Cette solution a été étudiée pour une prise de température au poignet et non via une caméra thermique à l'insu de son utilisateur. Les fonctionnalités de reconnaissance faciale ont été détournées pour vérifier le port d'un masque et le dispositif rendu autonome sans aucune autre connexion distante.

1. la détection du masque :

1-1 la famille YOLO en détection de masque et pourquoi Yolov4

Pour faire la détection du port du masque, On va traiter des algorithmes de langage python, Yolov4 et Darknet en temps réel pour détecter toutes les personnes qui vont passer utiliser les ascenseurs. [5]

En effet YOLOv4 est un modèle récent de détection d'objets en temps réel (avril 2020) qui a atteint des performances de pointe sur l'ensemble de données COCO. Il fonctionne en divisant la tâche de détection d'objets en deux parties, la régression pour identifier le positionnement de l'objet via des cadres de délimitation et la classification pour déterminer la classe de l'objet. Cette implémentation de YoloV4 utilise le Framework Darknet.

L'intérêt de Yolov4 dans les familles Yolo :

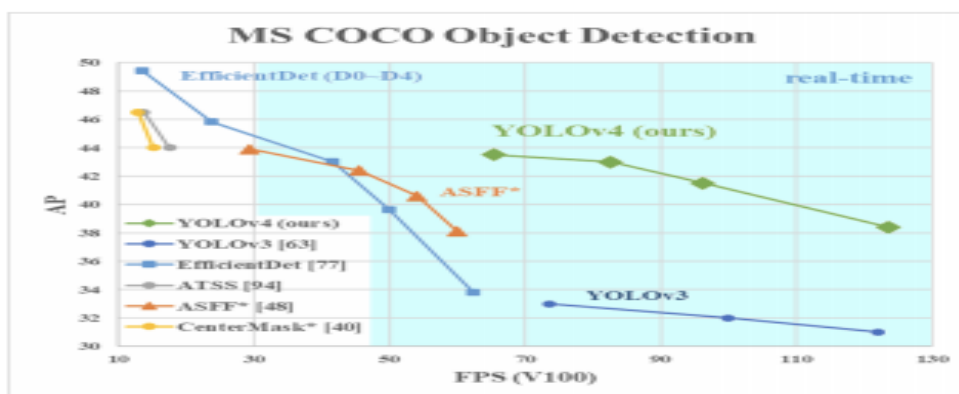


Figure 21 : YOLOv4 atteint l'état de l'art pour la détection d'objets

Dans cet article, nous examinons en profondeur les contributions à la recherche de YOLOv4 et les mettons dans le contexte de travaux antérieurs sur la détection d'objets. En résumé, YOLOv4 est une série d'ajouts de techniques de vision par ordinateur connues pour fonctionner avec quelques petites contributions novatrices. La contribution principale est de découvrir comment toutes ces techniques peuvent être combinées les unes avec les autres de manière efficace et efficiente pour la détection d'objets

1-2 Les étapes des algorithmes :

Etape 1 : générer les informations de base

Créer une base de données qui regroupe deux groupes, Des images d'êtres humains qui portent le masque de corona l'autre ou non , après avoir classé chaque image dans un groupe qui porte un code.



Figure 22 : les photos de la base de données

Etape 2 : lier la base de données à Google colab

- Insérer toutes les données à Google colab pour démarrer le Traitement.
- Cloner le dépôt darknet git sur la machine virtuelle colab
- Ensuite, créez un autre dossier nommé training dans le dossier yolov4-tiny. C'est ici que nous enregistrerons nos poids entraînés (Ce chemin est mentionné dans le fichier obj.data que nous téléchargerons plus tard)
- Exécutez le script python process.py pour créer les fichiers train.txt et test.txt dans le dossier de données.

Etapes 3 : faire l'entraînement de la machine

Lancer les algorithmes pour faire le Training à la machine

- Pour de meilleurs résultats, vous devez arrêter l'entraînement lorsque la perte moyenne est inférieure à 0,05 si possible ou au moins constamment inférieure à 0,3, sinon entraînez le modèle jusqu'à ce que la perte moyenne ne montre aucun changement significatif pendant un certain temps.[résultat d'excursion]

1-3 la description de démarches de programme dans la machine :

En tant que programmeurs, développeurs et praticiens de la vision par ordinateur/de l'apprentissage en profondeur, nous décrivons l'algorithme qui va gérer le travail de la détection :

- prendre des images normales de visages

-Créer ensuite un script Python de vision par ordinateur personnalisé pour leur ajouter des masques faciaux, créant ainsi un ensemble de données artificiel (mais toujours applicable dans le monde réel)

-Les repères faciaux nous permettent de déduire automatiquement l'emplacement des structures faciales, notamment :

*Yeux

*Les sourcils

*Nez

*Bouche

*Mâchoire

Pour utiliser des repères faciaux pour créer un ensemble de données de visages portant des masques faciaux, nous devons d'abord commencer par une image d'une personne ne portant pas de masque facial :

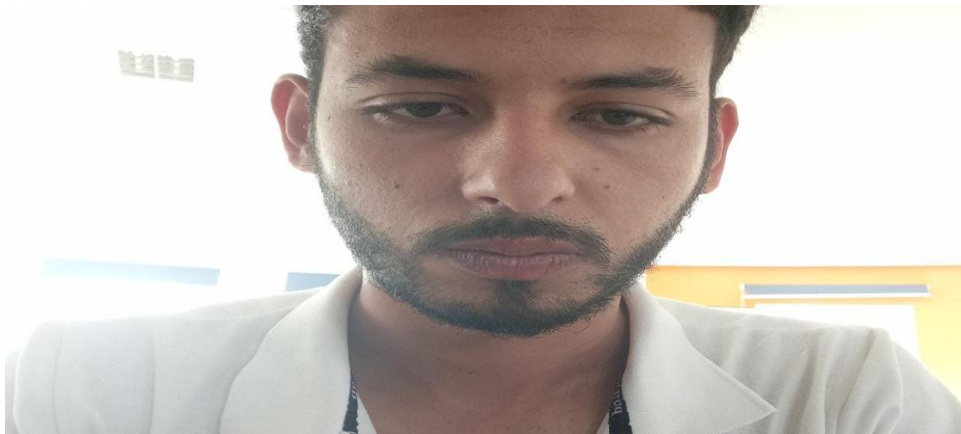


Figure 23: La photo sans masque

À partir de là, nous appliquons la détection de visage pour calculer l'emplacement de la boîte englobante du visage dans l'image :

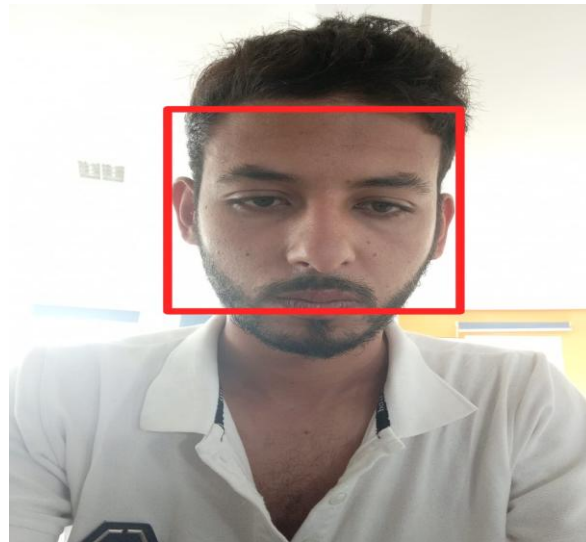


Figure 24: La photo avec cadre de détection

Fois que nous savons où se trouve le visage dans l'image, nous pouvons extraire la région d'intérêt (ROI) du visage :



Figure25 : les repères faciaux

Et à partir de là, nous appliquons des repères faciaux, nous permettant de localiser les yeux, le nez, la bouche, etc. :

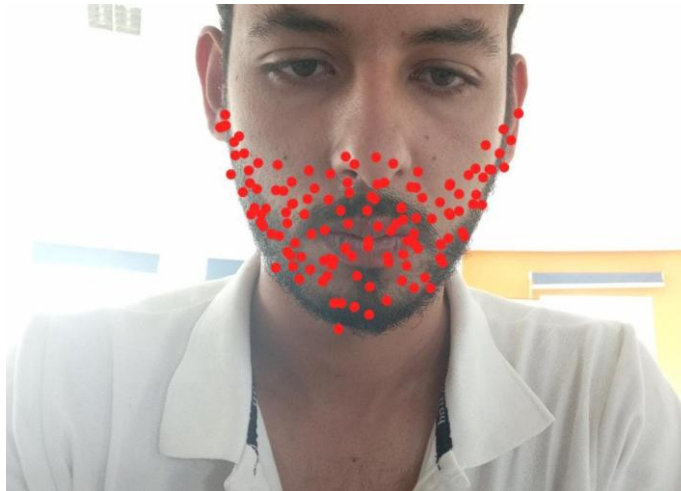


Figure26 : les repères faciaux à l'aide de dlib afin que nous sachions où placer un masque sur le visage.

Ensuite, nous avons besoin d'une image d'un masque (avec un fond transparent) tel que celui ci-dessous :



Figure 27: Un exemple de masque/écran facial COVID-19/Coronavirus.

Ce masque sera automatiquement appliqué sur le visage en utilisant les repères faciaux (à savoir les points le long du menton et du nez) pour calculer où le masque sera placé.

Le masque est ensuite redimensionné et pivoté, en le plaçant sur le visage :



Figure28 : photo de visage insérer avec masque de visage

Dans cette figure, le masque facial est placé sur le visage de la personne dans le cadre d'origine. Il est difficile de dire en un coup d'œil que le masque COVID-19 a été appliqué avec la vision par ordinateur au moyen de repères de visage OpenCV et dlib.

Nous pouvons ensuite répéter ce processus pour toutes nos images d'entrée, créant ainsi notre ensemble de données de masque facial artificiel :



Figure 29: photo de source de training

Figure : un ensemble artificiel d'images de masque facial COVID-19 est affiché. Cet ensemble fera partie de notre ensemble de données « avec masque » / « sans masque » pour la détection de masque facial COVID-19 avec vision par ordinateur et apprentissage en profondeur à l'aide de Python, OpenCV et TensorFlow/Keras.

L'implémentation a été faite sur Google Collaboration qui offre un GPU gratuit et très puissant (NVIDIA SMI : TESLA T4), après compilation on peut télécharger l'image ou la vidéo de l'output.

Les algorithmes :

```
# USAGE
# python detect_mask_image.py --image examples/example_01.png

# import the necessary packages
from tensorflow.keras.applications.mobilenet_v2 import preprocess_input
from tensorflow.keras.preprocessing.image import img_to_array
from tensorflow.keras.models import load_model
import numpy as np
import argparse
import cv2
import os
from google.colab.patches import cv2_imshow

# Load the input image from disk, clone it, and grab the image spatial
# dimensions
image = cv2.imread('/content/face-mask-detector/examples/example_02.png')
orig = image.copy()
(h, w) = image.shape[:2]

# construct a blob from the image
blob = cv2.dnn.blobFromImage(image, 1.0, (300, 300),
                              (104.0, 177.0, 123.0))

net=cv2.dnn.readNet('/content/face-mask-detector/face_detector/deploy.prototxt', '/content/face-mask-detector/face_detector/res10_300x300_
ssd_iter_140000.caffemodel')

# pass the blob through the network and obtain the face detections
print("[INFO] computing face detections...")
net.setInput(blob)
detections = net.forward()

# Loop over the detections
for i in range(0, detections.shape[2]):
    # extract the confidence (i.e., probability) associated with
    # the detection
    confidence = detections[0, 0, i, 2]
```

Figure30 : Les algorithmes de mask detection

Output image en temps réel :



Figure 31 : Output image mask detection

Output vidéo en temps réel :



Figure 32 : Output video mask detection

2.la mesure de la température :

La prise de la température corporelle permet de déceler l'apparition de fièvre. La fièvre est définie comme une élévation de la température interne du corps (température affichée lors d'une prise rectale) à plus de 38 ° C, en dehors de tout effort et dans une température ambiante tempérée. Elle n'est pas, en général, dangereuse en elle-même. Sa mesure est utile lorsqu'elle entraîne un inconfort et des symptômes.

Le contrôle de la température corporelle des gens est un outil qui indice la présence de coronavirus, la caméra thermique constitue une option complémentaire intéressante sous réserve d'en connaître les limites.

2-1 caméra thermique :

Pourquoi la détection de fièvre avec la thermographie ?

L'augmentation de la température corporelle (par rapport aux personnes des alentours), communément appelée « fièvre », est un indicateur important d'une infection. La thermographie est la méthode idéale pour scanner aussi bien des personnes individuelles que de grandes foules. La température est mesurée au coin intérieur de l'œil et une alarme est déclenchée si elle est hors des valeurs normales. Ainsi, on peut détecter rapidement et de manière fiable les personnes présentant une augmentation de la température corporelle afin de les séparer pour un contrôle plus approfondi.

Technologie infrarouge, IR pour contrôler la température corporelle

Tous les dispositifs utilisés doivent bien sûr être sans contact pour éviter toute transmission d'agent infectieux; C'est donc la **technologie infrarouge (IR)** qui est utilisée puisque les thermomètres infrarouges permettent de mesurer la température rapidement, à distance et sans toucher la personne ou l'objet que l'on veut tester.



Figure 33: image d une photo de caméra thermique

2-1-1 Détection des zones froides /chaudes.

La classification des pixels consiste à assigner une classe à chaque pixel en fonction de sa température. Un pixel peut être assigné à l'une des deux classes suivantes : pixel froid ou pixel conforme. Ces informations sont conservées en mémoire sous forme de segments. Un segment est un ensemble de pixels voisins appartenant à la même classe. Il est défini par une coordonnée de départ et une coordonnée d'arrivée. Un exemple de classification des pixels d'un profil de températures est donné dans la Figure 34. Les segments bleus sont formés des pixels froids et le segment vert contient le pixel conforme. Le segment rouge contient des pixels chauds. Étant donné que le lissage du profil de température a été effectué pour améliorer la détection de zones froides, ces pixels seront considérés comme étant conformes. Pour la détection de zones chaudes, le lissage avec un filtre d'ordre maximum doit être effectué. Ce seront alors les pixels froids qui seront considérés comme étant conformes.

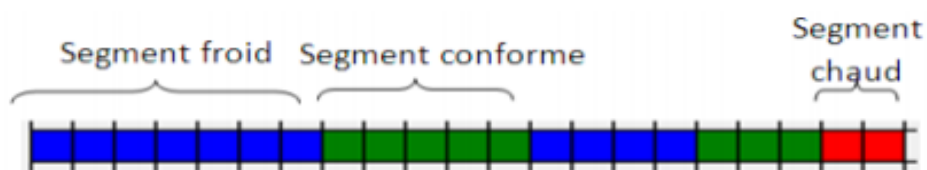


Figure34 : Classification des pixels en segments

2-3 capteur thermique :

Un capteur thermique de détection permet de mesurer la température des individus avec une grande précision. C'est un système particulièrement efficace pour repérer les personnes souffrant de fièvre et ainsi, lutter contre les épidémies.



Figure35: Capteur thermique et la résultat d'utilisation.

2-3-1 Principe de fonctionnement :

La longueur d'onde du rayonnement infrarouge dépend de la température. En analysant les rayonnements émis par le corps humain, un capteur thermique permet de détecter - à distance et en temps réel - la température corporelle d'un individu avec une très grande précision.

Avantages

- Rapidité et fluidité des contrôles (1 seconde par personne)
- Détection immédiate d'anomalie de température
- Réduction des risques d'infection

2-3-2 système de détection mobile :

Le contrôle s'effectue directement sur l'écran du capteur, et vous pouvez facilement vérifier la température de toutes les personnes se présentant devant elle. Avantages

- Système mobile, aisément déplaçable
- Jusqu'à 4h d'autonomie en fonctionnement continu
- Batteries rechargeables
- Précision de ± 0.5 °C, répondant au besoin de détection de fièvre préliminaire
- Solution livrée sur site et prête à l'emploi

3. Caméra thermique comme projet théorique .

Aujourd'hui toutes les caméras thermographiques sont des caméras matricielles et elles

Utilisent toutes un détecteur micro bolomètre au même titre que tous les PC utilisent un processeur

Pentium. Néanmoins, sur le marché de la thermographie infrarouge, toutes les caméras n'utilisent

pas le même détecteur et certains constructeurs vont même jusqu'à demander une fabrication

spécifique de détecteurs micro bolomètres selon leur propre cahier des charges. Le détecteur est le

principal responsable des caractéristiques techniques de la caméra en terme de sensibilité thermique

et de qualité d'image. Chaque fabricant de caméra thermographique propose par ailleurs des

logiciels spécifiques pour l'édition de rapports.

On peut profiter dans ce rapport dans notre projet smart ascenseur pour mesurer la température des visiteurs et faire une mesure et une détection de covid à partir la comparaison des résultats des rapports avec la température seuil 36.5 °C.

AUTREMENT DIT ? COMMENT ON PEUT REALISER NOTRE PROJET SMART ASCENSEUR A PARTIR LES CONCEPTES QU ON A ETUDIE ?

Chapitre 4 : Smart Ascenseur



Figure 36 : logo de projet

1.cahier des charges

Cahier des Charges Comme chaque projet nécessite un cahier de charge on a procédé dans un premier temps par établir un CDC (cahier des charges). Qui contient les différentes exigences :

Analyse du besoin :

Ce cahier de charge est une « sous partie » du cahier de charges du Smart Ascenseur. Le rôle principal du système est de protéger l'utilisateur contre le virus Corona soit par la détection du masque ou la mesure de la température de chaque utilisateur, ainsi la protection de la cabine de l'ascenseur contre le vol et le vandalisme. De plus traiter les images et les vidéos qui se trouvent dans le champ de notre Smart Ascenseur, afin de pouvoir aider les techniciens à prendre la bonne décision dans un temps minimal.(coté maintenance

1-1 bête a cornes :

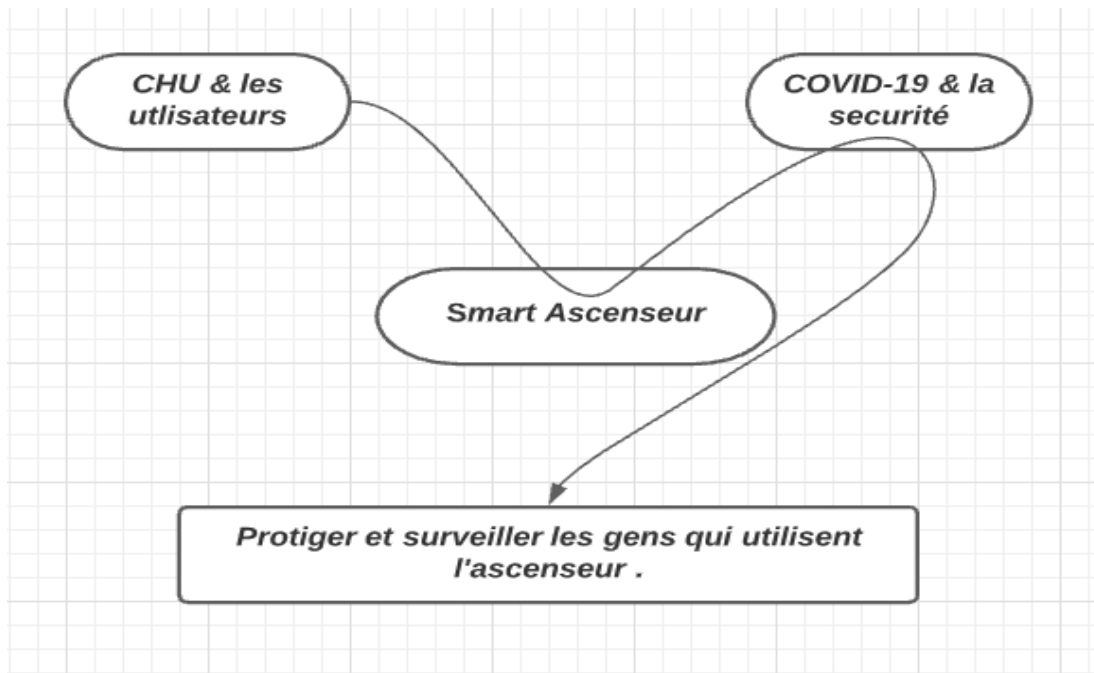


Figure37: bête a cornes de projet.

- Le système doit être capable à détecter les utilisateurs qui utilisent l'ascenseur soit à l'intérieur ou en face du l'ascenseur(demande de l'ascenseur). Quels que soit leurs comportement (en mouvement ou pas), leurs attitudes (de face, de coté), leur taille (grand, petit), leur nombre (seul ou en groupe) avec un taux de détection le plus maximal possible.
- Le projet doit être capable à collecter et analyser les données des caméras pour classifier les utilisateurs à température plus que 37°C (malades ou pas) ,ils portent le masque ou pas ,leur comportement (tranquilles ou insupportable) ,.....qui est détectés avec une précision maximale.
- L'ascenseur doit être capable à commander lui-même (bloquer, alerter, rendre...). Dans le cas où il y a des excitations des capteurs (Caméra thermique...)

1-2 Expression fonctionnelle du besoin :

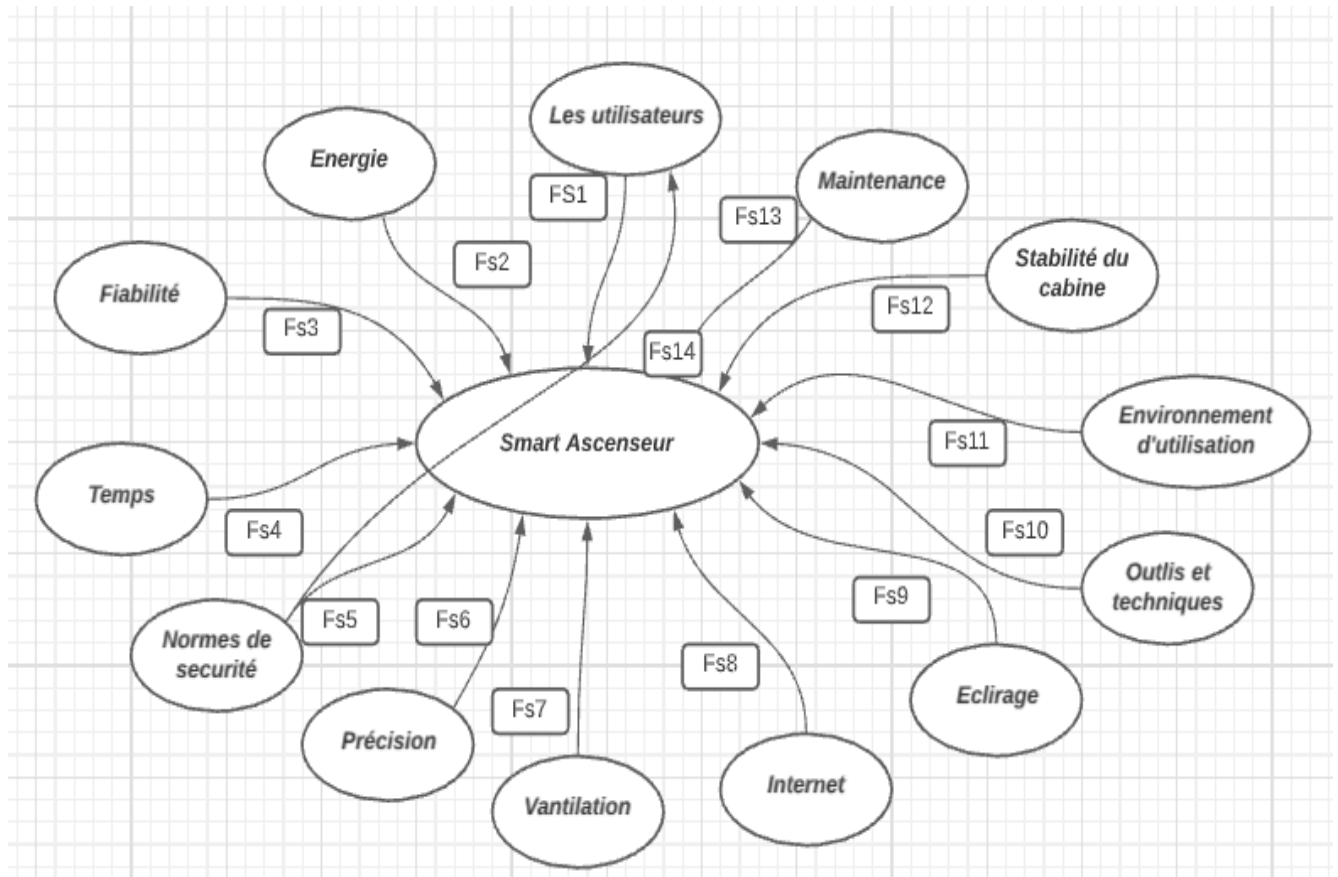


Figure 38: diagramme pieuvre de « Smart ascenseur »

1-3 Les normes :

Ne jamais mettre en service un ascenseur s'il n'a pas de contrat d'entretien ou un contrôleur.

Ne jamais ouvrir les portes ou continuer le travail d'ascenseur si il y a un vandalisme .

Pour l'éclairage de secours, préciser dans le CCTP un niveau d'éclairage minimum dans la cabine (20 lux minimum par exemple) ainsi que son autonomie (pendant 1 heure minimum par exemple) car la réglementation n'impose rien sur ces deux points.

La vitesse du cabine $v=1$ m/s .

Le nombres des caméras : 3 caméras ;

- L' un en haute de la porte palière ascenseur. Le type caméra digital
- Les autres au fond du cabine, l' un caméra thermique l autre caméra digital .

Départ cabine impossible si la température du local machine > 40 °c.

La boîte à bouton devra être accessible aux handicapés et comprendra :

- 1 bouton d'envoi par niveau desservi.
- 1 bouton de réouverture des portes.
- 1 bouton de fermeture des portes.

- Les flèches de direction cabine.
- L'affichage de la position de la cabine.

Des capteurs contre le vandalisme (sensible à les mouvement bizarre)

le bouton de blocage des portes ouvertes est interdit, mais un contact à clé est autorisé. Position de la boîte à boutons à l'intérieure de la cabine : à droite en entrant dans le cas d'un appareil à service simple à droite en entrant par le côté le plus usité dans le cas d'un ascenseur à double service.

Toujours demander un rapport par un bureau de contrôle (même sur un appareil neuf certifié CE).

Fonctions de services	Expression de la fonction de service	Fp	Fc
Fs1	Etre capable à détecter et servir les utilisateurs qui se trouvent dans l'ascenseur et l'entourage de l'ascenseur .x	x	
Fs2	Etre alimenté en énergie électrique et Consommer une énergie minimale		x
Fs3	Etre facile à utiliser par l'utilisateur .		x
Fs4	Réduire le temps d'exécution.		x
Fs5	Respecter les normes de sécurité .		x
Fs6	avoir une précision maximale lors de la détection et la classification.		x
Fs7	Etre capable de conserver la température au niveau de la météo extérieur.		x
Fs8	Lier à la connexion internet.		x
Fs9	Allumer avec un bon éclairage .		x
Fs10	Utiliser des outils et techniques adéquats répondant aux exigences imposées (AI, ML, Python et ses bibliothèques ...).		x
Fs11	S'intégrer à l'immeuble .		x
Fs12	Stabiliser la cabine.		x

Fs13	Prévoir une maintenance.		x
Fs14	Sentir le repos et la relaxation .		x

Figure39 : Tableau des fonctions de service.

2 Le système Raspberry :

Le Raspberry Pi est une série d'ordinateurs mono carte. Ils sont économiques, performants et de la taille d'une carte de crédit. Le Raspberry Pi a été développé au Royaume-Uni par la Fondation Raspberry Pi. L'objectif de la Fondation Raspberry Pi est de « faire progresser l'éducation des adultes et des enfants, en particulier dans le domaine de l'informatique, de l'informatique et des sujets connexes. » De nombreuses personnes ont utilisé Raspberry Pis pour fabriquer des choses comme des appareils photo, des consoles de jeux vidéo, des robots, serveurs Web et centres multimédias.

2-1 La série de Raspberry utilisé dans le projet :

Le Raspberry Pi 4 est de sortie, voici la fiche technique

Processeur 1.5GHz quad-core 64-bit ARM Cortex-A72.

1GB, 2GB, ou 4GB de mémoire SDRAM LPDDR4.

Ethernet Gigabit.

Wifi Dual-band 802.11ac.

Bluetooth 5.0.

Deux ports USB 3.0 et deux ports USB 2.0.

Support double écran.

GPU VideoCore VI.



Figure 40 : image de RaspBerry Pi 4

2-2 la fiche technique de caméra utilisée .

Le nom : testo 868 – Thermographier de façon intelligente et connectée.

Thermographie connectée – Voilà ce que vous permet la caméra thermique testo 868. Celle-ci présente la meilleure qualité d'image de sa catégorie, intègre un appareil photo numérique et convainc grâce à de nouvelles fonctions intelligentes. L'Application mobile Thermography intègre les valeurs de mesure sans fil et fait de votre Smartphone ou tablette un second écran pour votre caméra thermique. L'Application vous permet également de commander la caméra et de créer / envoyer des rapports directement sur site.

La fiche technique :

Débit d'images infrarouges	
Résolution infrarouge	160 x 120 pixels
Sensibilité thermique (NETD)	< 100 mK
Champ de vision / Distance de mise au point min.	31° x 23° / < 0,5 m
Résolution géométrique (IFOV)	3,4 mrad
testo SuperResolution (pixels / IFOV)	320 x 240 pixels / 2,1 mrad
Fréquence d'acquisition d'images	9 Hz
Mise au point	Mise au point fixe
Bande spectrale	7,5 ... 14 µm
Débit d'images visuel	
Taille de l'image / Distance de mise au point min.	min. 3,1 MP / 0,5 m
Représentation de l'image	
Écran	8,9 cm (3,5") TFT, QVGA (320 x 240 pixels)
Possibilités d'affichage	Image IR / Image réelle
Palette de couleurs	Fer, arc-en-ciel HC, froid-chaud, gris
Interfaces de données	
Connectivité WiFi	Communication avec l'Application mobile testo Thermography Module radio WiFi (UE, EFTA, USA, AUS, CDN, TR)
USB 2.0 Micro B	✓
Mesure	
Étendues de mesure	Étendue de mesure 1 : -30 ... +100 °C Étendue de mesure 2 : 0 ... +650 °C
Précision	± 2 °C, 2 % de la valeur de mesure

Stockage d'images	
Format de fichier	.bmt et .jpg ; exportation possible aux formats .bmp, .jpg, .png, .csv, .xls
Mémoire	Mémoire interne (2,8 GB)
Alimentation en courant	
Type d'accumulateur	Accumulateur au Lithium-Ion remplaçable sur site
Autonomie	4 heures
Options de chargement	Dans l'appareil / Sur la station de charge (option)
Fonctionnement sur secteur	✓
Conditions environnementales	
Température de fonctionnement	-15 ... +50 °C
Température de stockage	-30 ... +60 °C
Humidité de l'air	20 ... 80 %HR (sans rosée)
Indice de protection du boîtier (CEI 60529)	IP 54
Vibration (IEC 60068-2-6)	2G
Caractéristiques physiques	
Poids	510 g
Dimensions (LxIxP)	219 x 96 x 95 mm
Boîtier	PC - ABS
Logiciel PC	
Configurations requises	Windows 10, Windows 8, Windows 7
Normes, contrôles, garantie	
Directive UE	CEM : 2014/30/UE RED : 2014/53/UE
Garantie	2 ans

Figure 41 : tableau fiche technique du caméra thermique testo

3 Raspberry Pi 4 au service de Smart Ascenseur

Associant la mesure de température réalisée par un procédé infrarouge traditionnel à des techniques sophistiquées de reconnaissance faciale exploitant des algorithmes d'intelligence artificielle (IA), le système de contrôle d'accès de la gamme smart ascenseur se distingue par son efficacité et sa polyvalence. Il est parfaitement adapté aux ascenseurs.

Le système intelligent de contrôle d'accès par imagerie thermique de la gamme Raspberry Pi 4, qui se distingue par sa haute qualité de conception, est parfaitement adapté aux zones où les flux de personnes, le trafic et la sécurité doivent être particulièrement surveillés, en particulier au point d'entrée et de sortie de l'ascenseur. Il suffit que la personne qui souhaite d'utiliser l'ascenseur de le demander pour que la technologie intégrée procède rapidement au processus d'identification de l'individu et mesure sa température. Si la température

corporelle relevée dépasse la valeur normale, le système déclenche une alarme sonore et empêche toute autre entrée, afin de pouvoir réaliser un contrôle complémentaire. ainsi envoyer un message à la service de contrôle.

Ce système de contrôle d'accès de la gamme Raspberry Pi 4 est équipé d'une caméra thermique et caméra numérique d'un écran couleur de 7 pouces affichant des images avec une résolution de 1080P à 25 images par seconde qui permet de suivre et d'optimiser la détection des visages. Grâce à sa dynamique élevée, la caméra HD binoculaire fournit des images de reconnaissance faciale dans des environnements lumineux complexes en moins de 0,2 seconde. La technologie d'IA sophistiquée permet la reconnaissance rapide des individus avec une extrême précision, même s'ils portent des lunettes, des masques faciaux ou d'autres accessoires. La caméra thermique infrarouge est tout aussi précise. Elle peut mesurer des températures avec une précision allant jusqu'à $\pm 0,2$ °C.

De puissants outils logiciels permettent d'utiliser l'appareil pour réaliser, simultanément ou séparément, un dépistage sanitaire et/ou une reconnaissance faciale. Ce dispositif intègre une bibliothèque de reconnaissance faciale de 50 000 enregistrements et peut sauvegarder jusqu'à 100 000 registres d'identification. Le logiciel autonome (Python : YOLOv4) de gestion de la plate-forme Cloud permet la collecte automatique des visages et l'importation de liste par lots, ce qui offre un gain de temps et une mise en œuvre aisée. Il facilite également la gestion centralisée des données d'entrée telles que la température corporelle, les visiteurs et les registres d'entrée.

Son boîtier est conçu en polycarbonate ultra-résistant et . Il présente un indice IP54 de protection aux projections d'eau et de poussière et a été soumis à des tests de résistance à des chutes de 2 mètres. Le système est donc suffisamment robuste pour être installé à l'ascenseur .

IP54= L'indice de protection (IP) est une norme internationale de la Commission électrotechnique internationale relative à l'étanchéité ,la poussière, les bruits électriques .

4-Smart Ascenseur au service de CHU

- Associant la mesure de la sécurité réalisée par un procédé de détection des utilisateurs par des techniques sophistiquées de reconnaissance faciale exploitant des algorithmes d'intelligence artificielle,
- on a l'option de capter les comportements de vandalisme grâce à un capteur sensible fixé au boîtier à bouton.
- Dans le cas où l'utilisateur commet un sabotage et un vol, smart ascenseur arrête le mouvement entre deux étages et informe l'Unité de Sécurité et de Surveillance.
- Reclame les utilisateurs de porter les bavettes pour protéger l'environnement sanitaire de CHU , par un message vocal , dans le cas où il y a un utilisateur sans bavette .
- Un rappel à chaque période pour les responsables de faire la visite technique , la maintenance , et le contrôle.
- Un rapport statique de flux des gens et donner le pourcentage des gens qui portent le masque
- Protège les gens qui sont très importants

-envoyer des comptes rendu par messages soit à la boîte e-mail ou à la telephone mobile.

Les solutions d entretien des ascenseurs :

Les ascenseurs se sont le Centre névralgique de CHU, l'ascenseur fait face à un double enjeu. Il s'agit d'une part de rendre l'ascenseur existant plus sûr pour qu'il y ait moins de pannes et, d'autre part, de maîtriser les coûts liés à leur entretien, répartis au sein des bâtiments.

Comme solutions on propose :

- L'étalonnage des capteurs de l'ascenseur (capteurs photo électrique, vitesse, poids ...) chaque weekend
- Appliquer le projet smart ascenseur (rendre l'ascenseur intelligent en le dotant de IA).
- faire le change des pièces chaque 2 mois avant de tomber en panne.
- la maintenance prédictive permet de prioriser de façon objective les investissements et de les planifier dans le temps
- faire le nettoyage des portes pour éviter le blocage des portes.
- contrôler les flux des personnes avec des caméras pour éviter la surcharge qui bloque l'ascenseur.

Conclusion générale

L'intelligence artificielle consiste à concevoir des machines capables de penser. À notre cas, Smart ascenseur est un système révolutionnaire appliqué pour résoudre des problèmes techniques et qui peut être évolué encore plus dans le futur.

Avant tout, Ce système pourra donc s'avérer très utile dans le futur, et ce pour une question de confort et sécurité.

Ce système permet de contrôler entièrement un ascenseur depuis l'utilisation simple vers la sécurité et la surveillance tout se passant en temps réel. Le projet permet de rendre des messages à des responsables pour intervenir, ce qui permet par conséquent de contrôler le flux à distance (par exemple depuis son travail).

Ce stage m'a permis d'épanouir, aussi bien dans le plan professionnel qu'au niveau du personnel, mes connaissances en domaine du travail et d'application pratique du système mis en œuvre. C'était enrichissant du début à la fin, tel que ça m'a permis de me familiariser avec les différents services existants.

Références

[Google colab \[6\]](#)

[Wikipédia \[5\]](#)

[Le journal Aujourd'hui le Maroc la page 1 et 2 \[4\]](#)

<http://hmf.enseiht.fr/travaux/projnum/book/export/html/3558>

[https://blog.octo.com/classification-dimages-les-reseaux-de-neurones-convolutifs-en-toute-simplicité/\[3\]](https://blog.octo.com/classification-dimages-les-reseaux-de-neurones-convolutifs-en-toute-simplicité/[3])

[https://www.futura-sciences.com/tech/definitions/informatique-reseau-neuronal-601/ \[1\]](https://www.futura-sciences.com/tech/definitions/informatique-reseau-neuronal-601/)

[https://www.lebigdata.fr/reseau-de-neurones-artificiels-\[2\]](https://www.lebigdata.fr/reseau-de-neurones-artificiels-[2])

[definition#:~:text=Un%20r%C3%A9seau%20de%20neurones%20artificiels,la%20famille%20du%20Dee](#)

[p%20Learning.](#)

<https://www.actuia.com/contribution/jeancharlesrisch/segmentation-et-detection-dobjets-en-temps-reel-avec-tensorflow/>

<https://www.programmingsought.com/article/1068852005/>

<https://github.com/rafaelpadilla/Object-Detection-Metrics>

<https://intellica-ai.medium.com/a-comparative-study-of-custom-object-detection-algorithms-9e7ddf6e765e>

https://www.researchgate.net/publication/342570032_A_review_Comparison_of_performance_metrics_of

[pretrained models for object detection using the TensorFlow framework](#)

<https://jonathan-hui.medium.com/object-detection-speed-and-accuracy-comparison-faster-r-cnn-r-fcn-ssd->

[and-yolo-5425656ae359](#)

[Le CHU Hassan II de Fès, meilleur centre hospitalier maghrébin | Aujourd'hui le Maroc \(aujourd'hui.ma\)](#)