

الجمهورية الجزائرية الديمقراطية الشعبية

**République Algérienne Démocratique et Populaire**

Ministère de l'Enseignement  
Supérieur et de la Recherche  
Scientifique  
Université Akli Mohand Oulhadj-  
Bouira-  
Tasdawit Akli Muhand Oulhağ n  
-Tuviret-  
Faculté des Sciences et Sciences  
Appliquées



وزارة التعليم العالي و البحث العلمي  
جامعة اكلي محند اولحاج  
-البويرة-  
كلية العلوم و العلوم التطبيقية

**Mémoire de Master**

**Présenté au :**

**Département:** Génie Electrique

**Domaine:** sciences et technologie

**Filière:** Télécommunication

**Spécialité:** Systèmes des Télécommunications

**Réalisé par :**

SALHI Lahna

Et

DAHMANI Roza

**Thème:**

**Détection automatique de masque pour faire face au  
Covid-19**

Soutenu le 07/07/2022

Devant la commission composée de :

Mr. NOURINE Mourad	MCA	UAMOB	Président
Mme. KABLA Aida	MCB	UAMOB	Examinateur
Mr. CHELBI Salim	MCB	UAMOB	Encadreur

**Année Universitaire 2021/2022**



## *Dédicace*

*Au nom de Dieu, le Tout miséricordieux, le Très miséricordieux*

*Avec l'expression de ma reconnaissance, je dédie ce modeste travail à l'être le plus cher de ma vie, la femme qui n'a jamais cessé de me soutenir et de m'épauler pour que je puisse atteindre mes objectifs, ma source de vie, ma mère.*

*A mon précieux offre de Dieu, mon père à qui je dois ma réussite et tout mon respect, ta bienveillance me guide et ta présence à mes côtés a toujours été ma source de force.*

*A mes trois piliers dans cette vie, mes chers frères Amine, Belkacem et Mohand Ouamar, puisse Dieu vous donner santé bonheur et réussite.*

*A ma chère grand-mère qui a toujours été là pour nous tous, que Dieu la protège et lui offre la santé.*

*A mon fiancé Mohamed, mon confident, pour son soutien moral, sa patience et sa compréhension.*

*A mes meilleures amies Dounia, Soulaf, Assia, Sylvia, Amel, Sarah, Rosa, Thafeth, Nassima, Manel, Ryma et Tinhinane qui m'ont encouragé tout au long de ce projet. A toutes les personnes qui ont une précieuse place dans mon cœur.*

*Merci pour votre amour et soutien.*

*LAHNA*





## *Dédicace*

*Je dédie ce modeste travail :*

*A ma très chère mère YAMINA, Source inépuisable de tendresse, de patience et de sacrifice. Ta prière et ta Bénédiction m'ont été d'un grand secours tout au long de ma vie. Quoique je puisse dire et écrire, je ne pourrais exprimer ma grande affection et ma profonde reconnaissance. Puisse Dieu tout puissant, te préserver et t'accorder santé, longue vie et Bonheur.*

*A mon très cher père MOHAMMED AREZKI, Pourriez-vous trouver dans ce travail le fruit de toutes vos peines et tous de vos efforts. En ce jour. Aucune dédicace ne saurait exprimer mes respects, ma reconnaissance et mon profond amour. Puisse Dieu vous préserver et vous procurer santé et bonheur.*

*A mon frère KHALLED et ma belle-sœur MALIKA et mes sœurs SAMIA, NAIMA, AKILA, NADJOUA, AMINA, FARIDA, HANANE, Aucune dédicace ne peut exprimer la profondeur des sentiments fraternels et d'amour, d'attachement que j'éprouve à votre égard. Puisse dieu vous protéger, garder et renforcer notre fraternité.*

*A mes meilleures amies IMENE, LAHNA, PINOU, HINA, ILEA, qui ont été là pour moi, qui m'ont toujours soutenue et encouragé*

*A la mémoire de mes très chers grands-parents paternels et maternels.*

**ROZA**



## **Remerciements**

*En tout premier lieu, nous remercions Allah le tout puissant et miséricordieux de nous avoir donné le courage, la volonté et la santé pour mener à bien ce travail.*

*Nous voudrions exprimer nos sincères remerciements à notre promoteur « Mr. CHELBI Salim » pour sa qualité d'encadrement exceptionnelle, sa disponibilité permanente, sa patience et sa générosité en connaissances, grâce à laquelle nous avons pu enrichir notre savoir, nous sommes reconnaissantes à ses qualités humaines et pédagogique, à sa rigueur et à ses conseils si constructifs, sans lesquels, notre travail n'aurait pas été si riche et n'aurait pas vu le jour.*

*Nos vifs remerciements vont également aux membres de jury pour l'intérêt qu'ils ont porté à notre travail, en acceptant de l'examiner et de l'enrichir par leurs propositions.*

*Nous remercions aussi tous les enseignants qui ont contribué à notre formation, à qui nous exprimons notre profonde gratitude.*

*Enfin, nous tenons à remercier toutes les personnes qui ont participé de près ou de loin à la réalisation de ce travail.*

## Résumé

Le virus de la COVID-19 qui se propage rapidement dans le monde exige des mesures préventives de santé pour faire face. Ces dernières comprennent le respect des gestes barrière et le port obligatoire du masque dans les lieux publics. Dans ce travail, un modèle de détection automatique du port de masque sera introduit à l'aide d'un apprentissage. Le modèle proposé se comporte trois étapes, la première est la détection de visage en utilisant l'algorithme de Viola et Jones, ensuite dans la deuxième étape consiste à extraire les caractéristiques du visage, en utilisant séparément les deux descripteurs suivants : « HOG » et « LBP ». Une fois les caractéristiques extraites sont enregistrées, nous abordons la dernière étape qui est la classification, où nous allons appliquer le classifieur « SVM », pour la reconnaissance du port du masque.

**Mots-clés :** covid-19, port du masque, détection de visage, l'algorithme de Viola et Jones, HOG, LBP, SVM.

## Abstract

The COVID-19 Virus that is spreading almost all over the world requires preventative health measures to combat it. These include wearing a face mask in public places. In this thesis, a model will be proposed using "Deep Learning" which addressed the issue of face detection and mask wearing. This proposed model consists of three steps, the first of which is the face detection using the algorithm of "Viola and Jones", and then the second step which aims to extract facial characteristics; in this particular case we will be using the following descriptors: HOG and LBP. Once the extracted characteristics are recorded as matrices, we will start the third step which is devoted to classification, or we will apply the SVM classifier, for the purpose of recognition of mask wearing or the lack of it on a given person.

**Key words:** Covid -19. Wearing of mask, detection of the face, The algorithm of "Viola and Jones", HOG, LBP, SVM.

## ملخص

يتطلب فيروس كورونا الذي ينتشر في جميع أنحاء العالم تقريباً تدابير صحية وقائية لمكافحته. وتشمل هذه ارتداء قناع الوجه في الأماكن العامة. وفي هذه الأطروحة، سيتم اقتراح نموذج باستخدام «التعلم العميق» الذي يتناول مسألة كشف الوجه وارتداء الأقنعة. يتكون هذا النموذج المقترح من ثلاث خطوات، أولها اكتشاف الوجه باستخدام خوارزمية «فيولا وجونز»، ثم الخطوة الثانية التي تهدف إلى استخراج خصائص الوجه؛ في هذه الحالة بالذات سنستخدم الأوصاف التالية: HOG و LBP. بمجرد تسجيل الخصائص المستخرجة كمصفوفات، سنبدأ الخطوة الثالثة المخصصة للتصنيف، حيث سنطبق مصنف SVM، لغرض التعرف على ارتداء القناع أو عدم وجوده على شخص معين.

**الكلمات الرئيسية:** فيروس كورونا، قناع الوجه، كشف الوجه، خوارزمية «فيولا وجونز»، HOG, LBP, SVM

## Table des matières

<b>Introduction générale</b> .....	1
<b>Chapitre 01: Méthodes de détection de visage</b>	
1. Introduction.....	3
2. Détection de visage.....	3
3. Méthodes de détection de visage.....	4
3.1. Approche basée sur la connaissance.....	4
3.2. Approche basée sur les caractéristiques invariantes.....	5
3.2.1. La couleur de la peau .....	5
3.3. Approche basée sur l'appariement de gabarit (Template Matching) .....	6
3.4 Approche basée sur l'apparence.....	6
4. La méthode de Viola et Jones.....	7
4.1. Fonctionnement.....	7
4.1.1. Fonctionnalités de type Haar.....	8
4.1.2. L'image intégrale.....	9
4.1.3. Le principe.....	10
4.1.4. Sélection de caractéristiques par boosting.....	11
4.1.5. Cascade de classifieurs.....	12
5. Conclusion.....	13
<b>Chapitre 02: Extraction des caractéristiques du visage</b>	
1. Introduction.....	14
2. Motif binaire local (LBP: Local Binary Pattern).....	14
2.1. Concept de LBP.....	14
2.2. Etapes de la méthode LBP.....	15
3. SIFT (Scale Invariant Feature Transform) .....	16
3.1. Détection d'extrema d'espace échelle (scale-space) .....	16
3.2. Localisation des points d'intérêt.....	18
3.3. Assignment d'orientation.....	19
3.4. Descripteur de point-clé.....	20
4. Histogramme du gradient orienté.....	21
4.1. Les étapes d'implémentation du descripteur HOG .....	22
4.1.1. Normalisation de l'image de luminance.....	22

4.1.2. Calcul de l'image gradient.....	22
4.1.3. Calcul de l'histogramme.....	23
4.1.4. Formation de Normalisation des blocs.....	23
5. Classification.....	24
5.1. Classifieur KNN (K-nearest neighbors) .....	24
5.2. Classifieur SVM (Support vector machine) .....	25
6. Conclusion.....	25
<b>Chapitre 03: Application pour la détection de visage et port du masque</b>	
1. Introduction.....	26
2. Présentation d'outils de développement.....	26
2.1. Logiciel et langage d'implémentation .....	26
2.2. Environnement de travail.....	26
3. Présentation de l'application.....	26
4. Tests et résultats.....	27
4.1. Détection de visage.....	27
4.1.1. Test sur des images positives.....	27
4.1.2. Les différents cas de dysfonctionnement de l'algorithme.....	28
4.2. Extraction des caractéristiques.....	30
4.2.1 Extraction des caractéristiques par HOG.....	30
4.2.2 Extraction des caractéristiques par LBP.....	31
5. Classification.....	32
5.1. Création des bases de données pour l'entraînement.....	32
6. résultats et discussions.....	33
7. Conclusion.....	36
Conclusion générale.....	37
Bibliographie.....	38

*Liste des Figures*

<b>Figure 1.1.</b>	Détection du visage .....	3
<b>Figure 1.2.</b>	Différents types de méthodes de détection de visage .....	4
<b>Figure 1.3.</b>	Un visage typique de la méthode basée sur la connaissance .....	5
<b>Figure 1.4.</b>	La détection de la couleur de peau .....	5
<b>Figure 1.5.</b>	Processus de détection de la peau .....	6
<b>Figure 1.6.</b>	Exemple de caractéristiques pseudo-Haar .....	7
<b>Figure 1.7.</b>	Les trois types de caractéristique de type Haar que Viola et Jones ont identifié .....	8
<b>Figure 1.8.</b>	La représentation d'une image intégrale .....	10
<b>Figure 1.9.</b>	Sélection par Boosting .....	11
<b>Figure 1.10.</b>	Illustration de l'architecture de la Cascade .....	12
<hr/>		
<b>Figure 2.1.</b>	Un exemple de l'opérateur de base LBP .....	15
<b>Figure 2.2.</b>	Différence de gaussienne .....	18
<b>Figure 2.3.</b>	Construction de l'histogramme de l'orientation .....	20
<b>Figure 2.4.</b>	Construction d'un descripteur SIFT. ....	21
<b>Figure 2.5.</b>	Calcul de l'histogramme de gradient .....	23
<b>Figure 2.6.</b>	Normalisation des blocs .....	24
<hr/>		
<b>Figure 3.1.</b>	Diagramme de notre application .....	27
<b>Figure 3.2.</b>	Détection d'un seul ou plusieurs visages .....	27
<b>Figure 3.3.</b>	Problème d'inclinaison des visages lors de la détection .....	28
<b>Figure 3.4.</b>	Résultat de la détection des visages partiellement cachés .....	28
<b>Figure 3.5.</b>	Résultat dans le cas d'un visage avec des expressions faciales .....	29
<b>Figure 3.6.</b>	Résultat de la détection à différent éclairage .....	29
<b>Figure 3.7.</b>	Image montrant la détection des faux positifs. ....	30
<b>Figure 3.8.</b>	Détection et extraction des caractéristiques d'un visage avec et sans masque avec HOG ....	31
<b>Figure 3.9.</b>	Extraction des caractéristiques avec et sans masque avec le descripteur LBP .....	31
<b>Figure 3.10.</b>	Quelques images de la base de données des visages sans masque .....	32
<b>Figure 3.11.</b>	Quelques images de la base de données des visages avec masque .....	32
<b>Figure 3.12.</b>	Test du port ou non port du masque sur un seul visage .....	33



**Figure 3.13** : Test sur un visage portant un masque imprimé.....33  
**Figure 3.14** : Test sur un visage portant un masque transparent.....33  
**Figure 3.15**: Test du port ou non port du masque sur plusieurs visages.....34

*Liste des Tableaux*

**Tableaux 3.1.** Les performances de classification pour les deux bases de données avec le descripteur « HOG » ..... 35  
**Tableaux 3.2.** Les performances de classification pour les deux bases de données avec le descripteur « LBP»..... 35

## Liste des acronymes

**AdaBoost:** Adaptive Boosting

**DoG:** Difference of Gaussians

**HOG:** Histogram Of Oriented Gradients

**KNN:** K-Nearest Neighbors

**LBP:** Local Binary Pattern

**SIFT:** Scale Invariant Feature Transform

**SVM :** Support Vector Machine

# Introduction Générale

### INTRODUCTION GENERALE

La covid-19 est un type de la famille des coronavirus, qui apparaît le 16 novembre 2019 à Wuhan puis commençant à se propager jusqu'à ce qu'elle a pu encercler toute la planète. Ce virus peut être porté sur des toutes petites particules de sécrétions respiratoires (appelées aérosols) qui contaminent les personnes qui les inhalent. L'hygiène, le lavage régulier des mains, les désinfectants à base d'alcool, le maintien de la distance sociale, et le port du masque chirurgical réduit le risque de contamination. [1]

Pour la sécurisation des données, l'accès à des bâtiments, les systèmes de surveillance routiers et urbains, La question du contrôle automatisé du port du masque se pose. D'un point de vue technique, une véritable résolution a eu lieu au cours de ces dernières années, en particulier dans le domaine de la reconnaissance des objets par le deep Learning.

Les chercheurs ont démontré que pour reconnaître un visage, l'homme s'en serve de différentes caractéristiques qui varient selon la texture, la géométrie et les couleurs de différentes régions du visage tel que le nez, les yeux, la bouche... . Grace à cette remarque plusieurs études ont été mise au point pour déterminer s'il est possible de modéliser ce comportement de manière informatique.

Les premiers systèmes de détection de visages ont mis au point dans les années 70, ils sont efficaces dans peu d'applications, comme l'identification de photographie de passeport. La recherche et les premières méthodes se développent surtout à partir des années 1990, plusieurs techniques ont été mises en place grâce au progrès du codage vidéo et la nécessité de l'identification faciale. [2]

Une étape majeure a été faite en 2001 avec la publication de la méthode de Viola et Jones, la première à pouvoir détecter les visages en temps réel. Cette méthode est devenue un standard et a été adoptée et améliorée par plusieurs chercheurs. Cela remonte également à la décennie 2000 que les méthodes fondées sur l'apparence qui apprennent les modèles de visage à partir d'une série d'images d'apprentissage, se révèlent en général supérieurs à d'autres types d'approche.

En effet, la détection automatique des visages, qui est une technique qui identifie ou localise des visages humains dans des images numériques, est utilisé dans de nombreuses applications, ces dernières comprennent la reconnaissance faciale, le suivi du visage, l'analyse des expressions faciales, le système biométrique, le système de vidéosurveillance, et le système d'interaction homme-machine.

La détection faciale a pour but de déterminer la présence ou l'absence des visages sur une image, et d'indiquer s'ils sont présents, les positions de ces visages et leurs étendus.

C'est précisément dans ce contexte que ce modeste travail s'inscrit. Notre travail consiste à créer un système pour le traitement de l'image fondé sur l'apprentissage dont l'objectif est de détecter le visage, et ensuite la reconnaissance du port du masque ou non sur la personne.

Ce mémoire comprend trois chapitres, les deux premiers introduisent des notions théoriques, le dernier les valide par des tests et des expérimentations :

Le premier chapitre est dédié à l'état de l'art de la détection des visages, et les différentes approches utilisées pour accomplir cette tâche.

Le deuxième chapitre est consacré aux méthodes d'extraction des caractéristiques dans une image plus précisément qui contient des visages, Nous allons décrire trois méthodes : LBP, SIFT et HOG. Comme il aborde également une phase essentielle dans notre travail qui est la classification, en citant ainsi quelques exemples de classifieurs ; tel que « SVM » et « KNN ».

Le dernier chapitre est dédié à la partie de la réalisation de notre système, où nous allons présenter les bases de données utilisées durant notre travail, tout en utilisant les deux descripteurs « HOG » et « LBP » pour entraîner nos bases d'apprentissages, ensuite faire nos tests avec le classifieur « SVM ».Finalement nous présenterons les résultats expérimentaux obtenus.

Le mémoire se clôture par une conclusion en citant quelques perspectives.

# Chapitre 1

## Méthodes de détection de visage

## 1. Introduction

De nos jours la détection et la reconnaissance des visages jouent un rôle vital dans notre société actuelle en raison de son utilisation pour une large gamme d'applications (comme dans nos cameras mobiles ...) d'où la nécessité de développement de plusieurs techniques permettant la détection faciale.

Aujourd'hui, le monde numérique progresse avec le développement de la technologie, cependant la détection faciale est l'un des aspects les plus exigeants du traitement de l'image. La détection du visage est la première démarche vers des applications plus évoluées tels que la reconnaissance faciale, l'interaction homme-machine, la reconnaissance des expressions faciales, le vieillissement normal ou le sexe d'une personne, la variation du pose...et pour cela, de nombreuses méthodes de reconnaissance des formes ont été proposées pour détecter le visage humain dans les images et les vidéos.

Dans ce chapitre, nous allons présenter de différentes méthodes de détection de visage.

## 2. Détection de visage

La détection de visage est une technologie qui détermine l'emplacement et la taille des visages dans des images ou des vidéos. Elle détecte les traits du visage et ignore le reste. Au cours de ces dernières années, la détection de visage est devenue une thématique populaire de la recherche en vision par ordinateur [2]. Elle est indispensable dans plusieurs applications y compris la sécurité, la biométrie, la robotique, commande des interfaces homme-machine, la photographie, la conférence vidéo, les systèmes de surveillance... .



**Figure 1.1 :** Détection du visage [3]

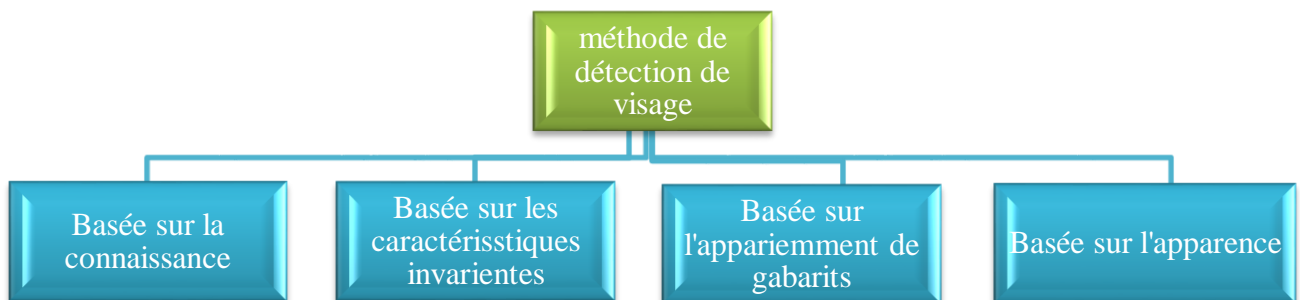
### 3. Méthodes de détection de visage

La première procédure pour la détection de visage s'appuie sur la capture d'image d'une vidéo ou sur une photo. Les traits du visage sont ensuite détectés et extraits selon la méthode choisie.

Dans ce qui suit, nous allons aborder différentes méthodes de détection de visage à partir d'une image.

On distingue généralement quatre grandes catégories :

- Approche basée sur les connaissances.
- Approche basée sur les caractéristiques invariantes
- Approche basée sur l'appariement de gabarits.
- Approche basée sur l'apparence.



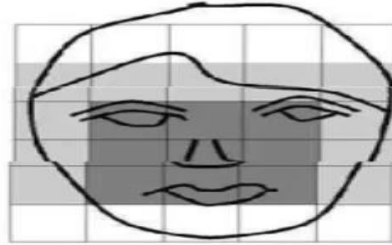
**Figure 1.2 :** Différents types de méthodes de détection de visage.

#### 3.1. Approche basée sur la connaissance

Le système détecteur exploite les caractéristiques standards qui forment une certaine signature numérique qu'il stocke dans une base de données, exemple : les visages sur les images ont souvent deux yeux symétriques, un nez et une bouche, distancés et positionnés les uns avec les autres. Les traits faciaux d'une image entrée sont extraits en premier, puis les candidats au visage sont reconnus. Cette méthode est conçue particulièrement pour la localisation du visage[4].



Les contraintes trouvées lors de l'application de cette approche sont non seulement la difficulté de créer un ensemble de règles strictes qui peuvent s'appliquer à toutes les formes sous différentes conditions, mais aussi la complexité de détecter le visage lorsqu'il est sur un arrière complexe. Cependant cette technique provoque plusieurs fausses détections.



**Figure 1.3 :** Un visage typique de la méthode basée sur la connaissance.

### 3.2. Approche basée sur les caractéristiques invariantes

Cette approche est dédiée à trouver les caractéristiques dans la vue d'exemple qui ne change pas, malgré la variation de position, d'éclairage ou d'angles de vision du visage.

L'inconvénient de cette méthode est que la qualité de l'image peut être réduite par l'illumination, le bruit ou l'occlusion. Nous citons deux méthodes appartenant à cette approche, méthode basée sur les caractéristiques faciale et la couleur de la peau.

#### 3.2.1. La couleur de la peau

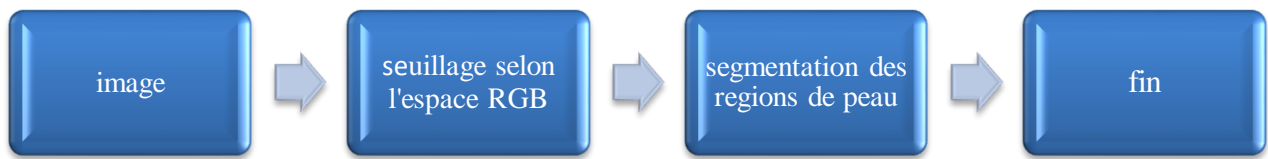
Cette méthode est utilisée lorsque les images en entrée sont en couleur, d'où la nécessité d'utilisation de cette caractéristique spécifique du visage pour mettre à part les zones susceptibles de contour des visages. [5]

Dans l'image, les pixels ayant la couleur de la peau seront détectés. Cependant, cette détection dépend principalement de l'espace couleur choisie (RGB ou Y Cb Cr...). Bien que la couleur de la peau diffère d'une personne à une autre, dans ce contexte plusieurs études ont prouvé que la couleur de la peau tenait plus la différence d'intensité que de la chrominance.



**Figure 1.4 :** La détection de la couleur de peau

Cette méthode se résume en trois étapes :



**Figure 1.5 :** Processus de détection de la peau.

### 3.3. Approche basée sur l'appariement de gabarits (Template Matching)

Cette approche est une position de processus de recherche de modèle de l'image à partir d'une image connue, qui est l'une des principaux contenus de la recherche sur la vision machine.

C'est à dire elle est utilisée pour obtenir les régions qui ont plus de possibilités d'être des visages humains[6], en détectant des visages entiers ou des parties du visage par le biais d'un échantillon standard d'apprentissage. La corrélation entre les images d'entrées et les exemples enregistrés est calculée et utilisée pour la décision.

### 3.4. Approche basée sur l'apparence

Ceci est une démarche statistique d'apprentissage automatique pour la détection faciale, elle utilise une série d'image contenant des visages pour entraîner le modèle. Ce modèle tire de cette série les particularités des visages. [7]

Les caractéristiques retirées sont exprimées sous la forme d'un modèle de distribution ou d'une fonction discriminante puis utilisées pour les tests. Dans le cas idéal, la détection doit être rapide et économique pour donner de bons résultats. Cependant ces méthodes demandent un long temps d'entraînement. Les méthodes appartenant à cette approche ont produit de bons résultats comparativement aux trois autres types de procédés. [8]

On peut citer parmi celles-ci:

- La méthode est basée sur les réseaux de neurones de Rowley et al.
- La méthode de Schneiderman et Kanade basée sur un classifieur de Bayes naïf.
- Eigen faces (visages propres).
- Algorithme de Viola et Jones.[Paul Viola and Michael Jones].

#### 4. La méthode de Viola et Jones

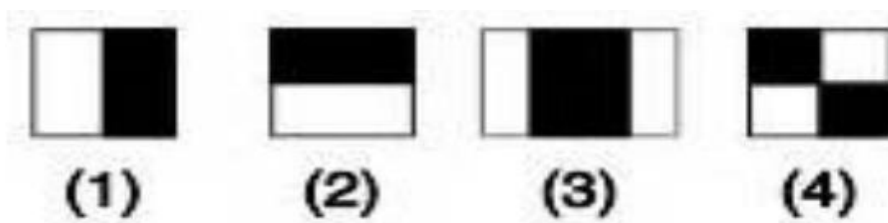
La méthode de Viola et Jones est une technique de détection d'objets basée sur l'apparence, publiée par Paul Viola et Michael Jones dans la revue scientifique IJCV en 2001 (Viola & Jones, 2001), permettant la détection d'objets dans les images numériques de manière robuste (taux de détection très élevé et taux de faux positifs très faible).

Consistant à parcourir l'ensemble de l'image en calculant un certain nombre de caractéristiques dans des zones rectangulaires qui se chevauchent, et cela en temps réel. À l'origine cet algorithme a été inventé pour détecter les visages humains, mais peut également être utilisé pour détecter d'autres types d'objets comme des voitures ou des avions...etc.

Ce cadre d'apprentissage supervisé se base essentiellement sur des caractéristiques pseudo-Haar [9], très peu coûteuses en calcul, la méthode d'apprentissage adaptatif de AdaBoost qui consiste à combiner plusieurs classifieurs faibles (ici les caractéristiques rectangulaires) afin de former un classifieur fort et enfin une architecture en cascade qui permet de stimuler le calcul.

##### 4.1. Fonctionnement :

Cette technique d'apprentissage automatique dédiée de base à la détection d'objets et en particulier de visage, elle permet d'analyser une image, sans avoir besoin d'étudier chaque pixel la constituant, et ce grâce à l'image intégrale qui se calcule à partir d'une image en utilisant quelques opérations par pixel, et cela permet de définir plusieurs zones rectangulaires au sein d'une image, ces zones quant à elles, permettent de calculer d'une manière rapide les caractéristiques appelées fonctionnalités; de type Haar qui sont en fait des masques permettant de déterminer plusieurs patterns utilisés par notre détecteur. [10]



**Figure 1.6 :** Exemple de caractéristiques pseudo-Haar [10]

Par exemple:

- le motif (2) exprime la variation de l'intensité de la lumière entre les yeux et le nez.
- le motif (3) exprime la variation de l'intensité de la lumière entre les yeux et les pommettes.

Ensuite, deux autres mécanismes s'imposent: La méthode d'apprentissage adaptative AdaBoost cherchant à minimiser l'erreur de classification, et l'algorithme en cascade de classifieur, servant à détecter la présence de visage sur une image.

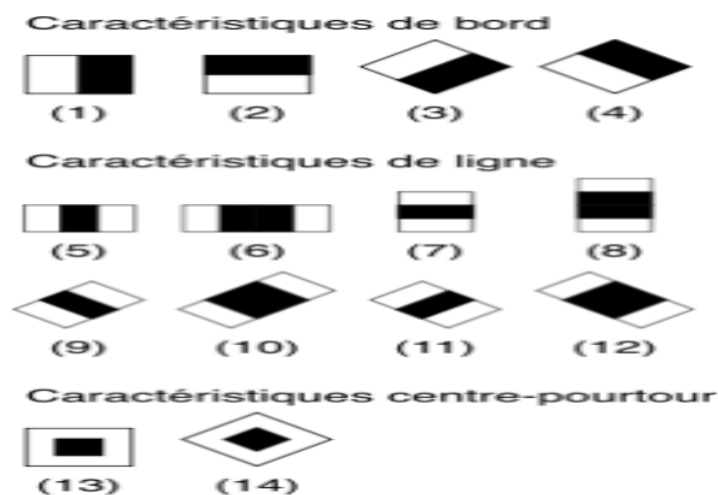
#### 4.1.1. Fonctionnalités de type Haar

Au 19ème siècle, un mathématicien hongrois Alfred Haar a introduit le concept d'ondelettes de Haar, qui sont une série de fonctions "au carré" mises à l'échelle formant un ensemble, une famille ou une base d'ondelettes.

Viola et Jones ont opté pour l'utilisation des ondelettes de Haar et ont développé les fonctionnalités dites du type Haar.

Les caractéristiques de type haar sont des caractéristiques utilisées en vision par ordinateur pour la détection d'objet dans des images numériques permettant de connaître la différence de contraste entre plusieurs régions rectangulaires adjacentes dans une image déjà convertit en niveaux de gris (car la différence de couleur n'est pas requise lors de la détection faciale), les intensités des pixels de ces régions sont additionnées, formant des sommes dont la différence constitue une caractéristique et seraient calculées en soustrayant la somme des pixels noirs à la somme des pixels blancs.

Il existe 3 types de caractéristiques du type Haar que Viola et Jones ont identifié dans leurs recherches :



**Figure 1.7 :** les 3 types de caractéristiques du type Haar que Viola et Jones ont identifié dans leurs recherches. [11]

Si vous regardez attentivement, les caractéristiques de Haar nous permettent d'identifier les bords de l'image en niveaux de gris. Chaque fois que le niveau d'intensité diminue ou augmente, il y a des bords, et ces caractéristiques nous permettent de les localiser.

Comme il y a de très nombreuses caractéristiques sur un visage, la détection de celles-ci, entraîne l'utilisation des centaines de variantes de caractéristiques Haar, ce qui est Malheureusement très coûteux en termes de ressources processeur, c'est là qu'interviennent les images intégrales

#### 4.1.2. L'image intégrale :

Pour calculer rapidement et efficacement ces caractéristiques sur une image, les auteurs proposent également une nouvelle méthode, qu'ils appellent image intégrale.

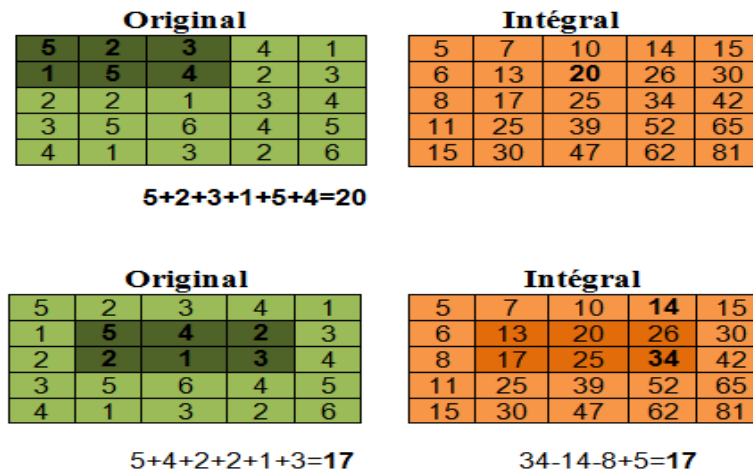
C'est une représentation sous la forme d'une image, de même taille que l'image d'origine, elle contient en chacun de ses points la somme des pixels situés au-dessus et à gauche du pixel courant. Plus formellement, l'image intégrale  $ii$  au point  $(x, y)$  est définie à partir de l'image  $i$  par [12]:

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \quad (1.1)$$

Le calcul de la somme des valeurs des pixels appartenant à une zone rectangulaire s'effectue donc en accédant seulement à quatre pixels de l'image intégrale : soit un rectangle ABCD dont les sommets sont nommés dans le sens des aiguilles d'une montre en commençant par le sommet supérieur gauche et soit  $x$  la valeur sous la représentation intégrale d'un sommet  $X$  du rectangle.

( $X \in \{A, B, C, D\}$ ). La somme des valeurs des pixels appartenant à ABCD est, quelle que soit sa taille, donnée par  $C - B - D + A$  [8].

Une caractéristique de Haar étant une combinaison linéaire de tels rectangles ABCD, son calcul se fait alors en un temps indépendant de sa taille. [10]



**Figure 1.8** : La représentation d'une image intégrale

Pour obtenir les valeurs de  $(x, y)$  de l'image intégrale, on doit faire la somme des pixels au-dessus et à gauche du pixel  $(x, y)$ . Ainsi, le calcul de la somme des valeurs des pixels appartenant à une région rectangulaire se fait en accédant uniquement aux quatre pixels de l'image intégrale : soit un rectangle ABCD avec des sommets nommés dans le sens des aiguilles de l'horloge, en commençant par le sommet supérieur gauche et soit  $x$  la valeur sous la représentation intégrale du sommet X du rectangle

( $X \in \{A, B, C, D\}$ ). La somme des valeurs des pixels appartenant à ABCD est, quelle que soit sa taille, donnée par  $C-B-D+A$ . [10].

#### 4.1.3. Principe

La méthode de Viola et Jones fonctionne en balayant une image tout en servant d'une fenêtre de détection de taille initiale 24px par 24px (dans l'algorithme original) qui génère environ 160 000 caractéristiques, mais seules quelques-unes sont importantes pour identifier et déterminer la présence d'un visage. Une fois que l'image est parcourue entièrement, la taille de la fenêtre est augmentée et le balayage recommence, jusqu'à ce que la fenêtre soit de même taille que l'image. Cette augmentation se fait par un facteur multiplicatif de 1.25, concernant le balayage, il permet simplement de décaler la fenêtre d'un pixel, ce décalage peut être changé afin d'accélérer le processus, mais un décalage d'un pixel assure une précision maximale. [10]

Cette méthode consiste à calculer ces caractéristiques à toutes les positions et à toutes les échelles dans la fenêtre de détection, dans des zones rectangulaires qui se chevauchent.

#### 4.1.4. Sélection de caractéristiques par boosting

Une fois les caractéristiques sont définies, la sélection de celles qui sont les plus discriminantes pour l'objet à détecter s'impose, d'où l'utilisation de l'algorithme de boosting.

L'algorithme de boosting n'est qu'une version adaptée d'AdaBoost, visant à la fois à la sélection et à combiner plusieurs caractéristiques (classifieurs faibles) pour former un classifieur très efficace (classifieur fort). Un classifieur faible est assurément un classifieur qui fonctionne mal, c'est à dire, il assure en moyenne une réponse meilleure qu'un tirage aléatoire. L'apprentissage du classifieur faible consiste alors à trouver la valeur seuil de la caractéristique qui permet de mieux séparer les exemples positifs des exemples négatifs. Le classifieur se réduit alors à un couple (caractéristique, seuil).

Le boosting est une branche de l'intelligence artificielle que Viola et Jones ont adapté en 2001 pour la détection de visage pour la première fois, ayant comme idée clé de sélectionner à chaque itération une caractéristique qui sera directement ajoutée à la liste des caractéristiques déjà sélectionnées aux itérations précédentes, et tout cela va contribuer à la formation d'un classifieur final assez fort. Cette sélection se fait en entraînant un classifieur faible pour toutes les caractéristiques et en sélectionnant celui avec l'erreur la plus faible sur l'ensemble d'apprentissage.

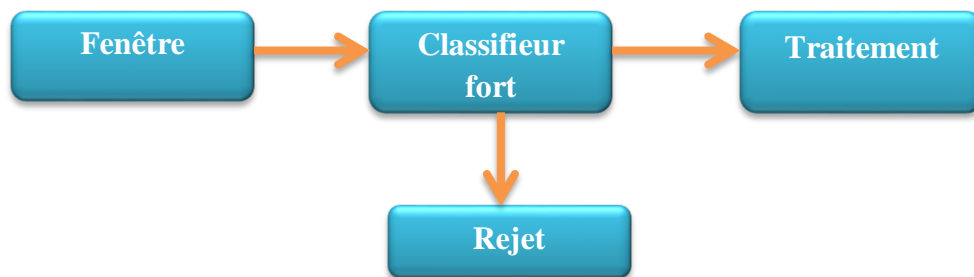


Figure 1.9 : Sélection par boosting.

Cet algorithme tient à définir à chacune de ses étapes, une nouvelle distribution de probabilité a priori sur les exemples d'apprentissages en fonction des résultats de l'étape précédente. [13]

A chaque étape 't', l'apprenant cherche une hypothèse 'h<sub>t</sub>' qui atténue l'erreur de classification et à la fin de cet algorithme, chaque règle de classification 'h<sub>t</sub>' est pondérée par une valeur α<sub>t</sub> calculée en cours de route. La classification d'un nouvel exemple se fait en suivant cette règle :

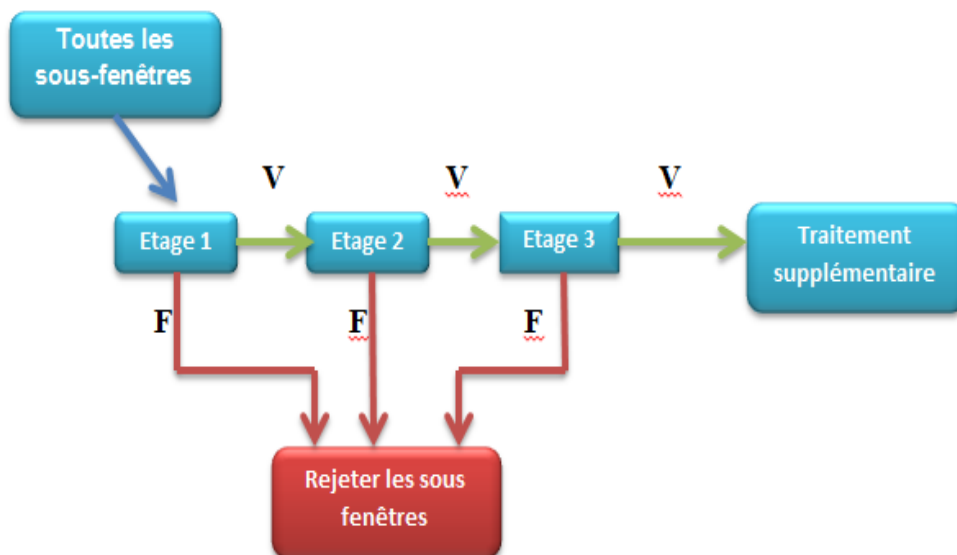
$$h(x) = f(x) = \begin{cases} 1 & \text{si } \sum_{i=0}^t \alpha_i h_i \geq \text{Seuil} \\ 0 & \text{sinon} \end{cases} \quad (1.2)$$

#### 4.1.5. Cascade de classifieurs

La méthode de Viola et Jones s'appuie sur une méthode de recherche exhaustive de l'image complète, qui vérifie la présence d'objets dans toutes les positions et sur différentes échelles de la fenêtre. Toutes fois le temps de calcul exigé par cette méthode reste considérable. Ainsi pour le réduire, il est essentiel d'organiser l'algorithme de détection dans le classifieur en cascade.

Cette structure en cascade peut également s'interpréter comme un arbre de décision dégénéré dans lequel, chaque étape est entraînée pour détecter autant d'objets intéressants que possible tout en rejetant une certaine fraction d'objets non-intéressants. [14]

Il est constitué de plusieurs couches, chacune étant la combinaison par Boosting de caractéristiques rectangulaires.



**Figure 1.10 :** Illustration de l'architecture de la cascade. [10]

La décision « fausse » met fin au calcul et exige pour le détecteur de se retourner. La décision « Vrai » transmet l'image d'entrée au suivant classifieur dans la cascade. Si les classifieurs ont voté « vrais », alors l'entrée est classifiée comme un vrai exemple (il y a un visage) et si l'un des classifieurs vote « Faux », le calcul s'arrête et l'entrée sera classifiée comme « Fausse ».

En réalité, la cascade est constituée d'une succession d'étages, dont chacun est composé d'un classifieur fort appris par AdaBoost. L'apprentissage du classifieur de l'étage  $n$  s'effectue à l'aide des exemples qui ont passé l'étage  $n-1$ ; ce classificateur doit donc faire face au problème plus compliqué : plus les étages sont hauts, plus les classifieurs deviennent complexes. [10]



Pour obtenir les taux de détection et les taux de fausses alarmes en fin de cascade; le choix du nombre  $K$  d'étages est déterminé par l'utilisateur, Viola et Jones utilisent  $K = 32$  étages. L'utilisateur doit également préciser le taux minimum de détection ' $d_i$ ' et le taux maximum de fausses alarmes ' $f_i$ ' si celui-ci doit être atteint pour l'étape ' $i$ '. Le taux de détection de cascade est ensuite déterminé par :

$$D = \prod_{i=1}^k d_i \quad (1.3)$$

Et le taux de fausse alarme par :

$$F = \prod_{i=1}^k f_i \quad (1.4)$$

Dans les faits, les taux  $d_i$  et  $f_i$  sont identiques pour tous les étages. De façon indirecte, ces Taux déterminent aussi le nombre de caractéristiques employées par les classifieurs forts à chaque étage : les itérations AdaBoost se poursuivent jusqu'à l'obtention du taux de fausses alarmes cible, de faibles caractéristiques ou classifieurs sont ajoutés jusqu'à l'atteinte des taux cibles, puis transférés à la prochaine étape.

## 5. Conclusion

Dans ce chapitre nous avons procédé à l'étude de la détection de visage, en présentant en vue générale ses différentes méthodes et leurs principes de fonctionnement. En effet, ces méthodes se divisent en quatre grandes catégories: méthode basée sur la connaissance, les caractéristiques invariantes, Template Matching et enfin la méthode basée sur l'apparence dans laquelle on a pris l'exemple de la méthode de viola et Jones pour la détailler.

Dans le prochain chapitre, nous allons présenter quelques méthodes d'extraction des caractéristiques du visage et classification.

# **Chapitre 2**

Extraction des caractéristiques  
du visage et classification

## 1. Introduction

L'extraction des caractéristiques faciales, est un processus très important afin d'augmenter les performances de localisation et le taux de détection du visage. En effet, on extrait de l'image les différentes informations et particularités qui vont être ensuite mémorisées pour être utilisées plus tard dans la phase de décision et cela se fait après avoir localisé le visage dans une image.

Dans ce chapitre, nous allons expliquer quelques méthodes d'extraction des caractéristiques à savoir, la méthode du motif binaire locale LBP, la méthode de « Scale-invariant feature transform » SIFT et la méthode basée sur l'histogramme du gradient orienté HOG. Ensuite nous allons aborder une phase très essentielle dans la détection de visage qui est la classification, citant son intérêt dans notre processus ainsi que ses différentes approches.

## 2. Motif binaire locale (LBP: Local Binary Pattern)

LBP a été d'abord mentionné pour la première fois par Harwood en 1993 afin de mesurer le contraste local d'une image mono spectrale, ensuite popularisé en 1996 par Timo Ojala pour analyser les textures [15].

Cet opérateur invariant aux variations d'intensités est devenu une approche populaire dans de nombreux domaines d'applications comme la reconnaissance et l'analyse faciales, l'analyse des textures, la restauration d'images, l'analyse de scène...et ce grâce à sa simplicité dans les calculs, son potentiel pour la reconnaissance de visage et sa robustesse face aux changements de pose et d'éclairage. Il est considéré le plus rapide de tous les algorithmes d'extraction de caractéristiques; cependant son application s'avère difficile en temps réel si le nombre de pixels voisin considéré est important.

### 2.1. Concept de LBP

Cet opérateur se base sur la division de l'image en plusieurs petites zones à partir desquelles les histogrammes des LBP uniformes sont extraits et concaténés en un seul vecteur de caractéristiques afin de représenter efficacement l'image du visage. Son concept consiste à en prendre en entrée un carré de 9 pixels, seuiller pour chaque pixel ses huit voisins directs avec un seuil dont la valeur est le niveau de gris du pixel courant. Tous les voisins prendront une valeur 1 s'ils sont supérieurs ou égaux au pixel courant et 0 s'ils sont inférieurs.

Le code LBP du pixel courant est obtenu concaténant les huit valeurs pour former un code binaire, en le lisant dans le sens des aiguilles d'une montre autour du pixel central, puis ce motif de bits est converti en valeur décimale représentant le nouveau code LBP pour le pixel central.

La conversion de toute une image se fait en déplaçant une fenêtre de conversion de 3×3 pixels sur toute l'image.

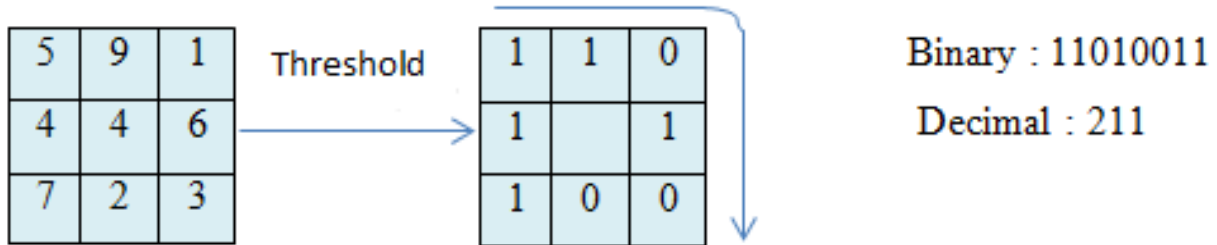


Figure 2.1: Un exemple de l'opérateur de base LBP.

La valeur LBP d'un échantillon d'image 3 × 3 est calculée comme suit :

$$LBP_{p,r} = \sum_{p=0}^{p-1} s(g_p - g_c)2^p \tag{2.1}$$

$$S(x) = f(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{sinon} \end{cases} \tag{2.2}$$

$g_c$  est la valeur de niveau de gris du pixel central

$g_p$  est la valeur de gris de ses voisins autour de  $g_c$

P est le nombre de voisins

Au final on obtient une image dont ses pixels ont une intensité qui se situe entre 0 et 255. [16]

### 2.2. Etapes de la méthode LBP

Après avoir converti l'image en espace en niveaux de gris. On prend une partie de cette image sous forme d'une fenêtre de 3x3 pixels (une matrice 3x3) dont chaque pixel a l'intensité entre [0 - 255]. Ensuite, nous prenons la valeur centrale de la matrice à utiliser comme seuil c'est à dire la valeur qui se situe au centre, Cette valeur sera utilisée pour définir les nouvelles valeurs des 8 voisins. Pour chaque voisin de la valeur centrale (seuil), nous définissons une nouvelle valeur binaire. Nous mettons 1 pour les valeurs qui sont égales ou supérieures au seuil et 0 pour les valeurs inférieures au seuil.

Désormais, la matrice ne contiendra que des valeurs binaires (en ignorant la valeur centrale). Nous faisons en sorte de concaténer chaque valeur binaire de chaque position de la matrice ligne par ligne dans une nouvelle valeur binaire (par exemple 10001101), à savoir qu'il existe d'autres approches pour concaténer les valeurs binaires, et elles donnent toutes le même résultat final.

Ensuite, nous convertissons cette valeur binaire en une valeur décimale pour la définir sur la valeur centrale de la matrice, qui est en fait un pixel de l'image d'origine. A la fin de cette procédure (procédure LBP), nous aurons une nouvelle image qui représente mieux les caractéristiques de l'image originale.

### 3. Scale Invariant Feature Transform (SIFT)

SIFT (Scale-invariant feature transform) ou bien «"transformation de caractéristiques visuelles invariante à l'échelle», est une approche utilisée dans le domaine de la vision par ordinateur proposée et développée en 1999 par le chercheur David Lowe. [17] [18]

Cette approche a été nommée «Scale Invariant Feature Transform» parce que la méthode transforme les données d'image en coordonnées invariantes sur l'échelle par rapport aux caractéristiques locales, elle permet de détecter et identifier les éléments similaires entre différentes images numériques. Cette approche transforme une image en un grand groupe de caractéristiques, dont chacune est invariante aux transformations suivantes : translation, changement d'échelle, rotation, et partiellement invariante aux changements d'illumination et projection affine ou 3D.

Lowe propose une approche en deux parties :

- Un algorithme pour la détection des caractéristiques et de calcul de descripteurs ;
- Un algorithme pour la mise en correspondance.

De ces deux aspects, le premier est certainement l'aspect le plus assuré de la popularité de la méthode de Lowe [19], et elle s'effectue en 4 étapes essentielles

- Détection d'extrema d'espace-échelle (scale-space).
- Localisation des points d'intérêt.
- Choix de l'orientation des descripteurs.
- Descripteurs de point-clé.

#### 3.1. Détection d'extrema d'espace-échelle (scale-space)

La détection d'extrema se réalise dans un espace discret appelé « espace des échelles » ou bien « scale space en anglais » qui contient trois dimensions (les coordonnées cartésiennes  $x$  et  $y$  et le facteur d'échelle  $\sigma$ ). Etant donné une image  $I$  et un noyau gaussien  $G$  de paramètre d'échelle  $\sigma$ , on appelle le gradient de facteur d'échelle (noté  $L$ ) qui est produit à partir de la convolution de  $I$  par  $G$  et il est donné par :

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (2.3)$$

Avec :

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (2.4)$$

La convolution de l'image originale par le filtre gaussien permet de lisser l'image  $I$  de manière à ce que même de très petits détails soient estompés. En conséquence, la détection des extrema dans l'espace des échelles se produit à l'aide de la fonction de Différence de Gaussienne «DoG»

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma) \end{aligned} \quad (2.5)$$

Où  $k$  est un paramètre constant qui dépend de la finesse de la discrétisation de l'espace des échelles voulue.

Il est recommandé d'utiliser la pyramide pour améliorer le temps de calcul des images floutées sur un grand nombre d'échelles différentes. Lowe utilise une pyramide dont la base est l'image originale et chaque niveau (octave) du niveau précédent est obtenu en divisant la résolution de l'image par 2, ce qui signifie doubler le facteur d'échelle. Le nombre d'intervalles par octave est défini par  $k$ . (voir la figure 2.2.)

Pour chaque octave d'espace d'échelle, l'image initiale est convolulée successivement par une Gaussienne pour produire des images différentes à échelle. La différence entre les images avec une échelle successive est calculée afin de produire la différence des Gaussiennes « DoG » correspondant comme indiqué à droite. Après chaque octave, l'image gaussienne est sous échantillonnée avec un facteur de 2, et le processus recommence. [16]

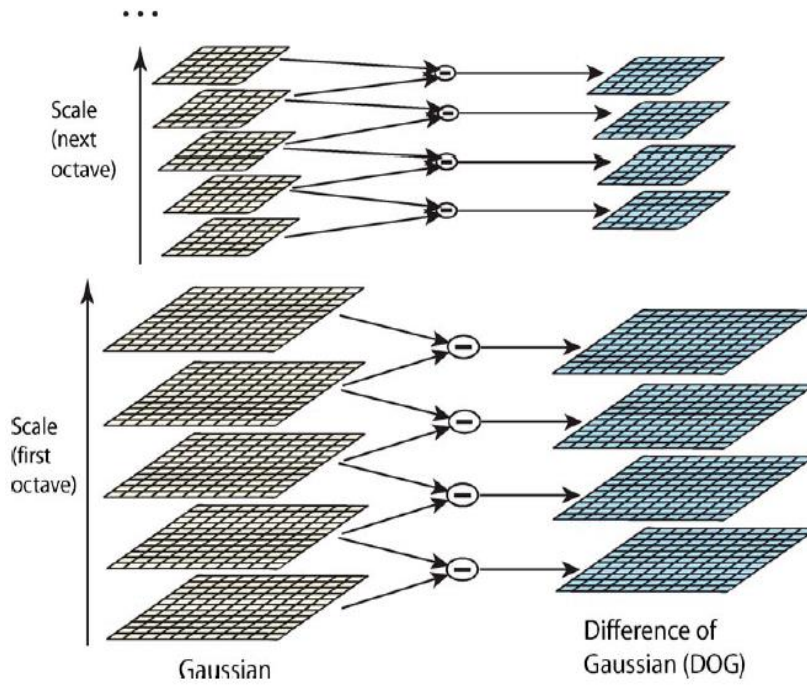


Figure 2. 2: Différence de Gaussienne. [16]

### 3.2. Localisation des points d'intérêt

L'interpolation de la fonction  $D(x, y, \sigma)$  vise à : premièrement, améliorer la position des points d'intérêt pour obtenir l'emplacement plus précis dans l'espace et à l'échelle. Deuxièmement, pour rejeter les points peu contrastés où sont situés sur des arêtes de contour à faible courbure .Il est donc susceptible de glisser facilement.

Cette interpolation s'obtient par le développement de Taylor de la fonction d'échelle-espace,  $D(x, y, \hat{y})$ , autour du point-clé candidat, et nous obtenons :

$$D(x) = D + \frac{\partial D^T}{\partial X} X + \frac{1}{2} X^T \frac{\partial^2 D}{\partial X^2} X \tag{2.6}$$

Où  $D$  et ses dérivées sont évalués au point-clé candidat et où  $X = (x, y, \sigma)^T$  est un décalage par rapport à ce point. La position exacte de l'extremum  $\hat{X}$  est déterminée l'équation annulant la dérivée de cette fonction par rapport à  $X$ ; on trouve ainsi :

$$\hat{X} = - \left( \frac{\partial^2 D}{\partial X^2} \right)^{-1} \frac{\partial D}{\partial X} \tag{2.7}$$

La valeur de la fonction à l'extrema,  $D(\hat{X})$ , est utile pour rejeter les extrema instables avec un faible contraste. Ceci peut être obtenu en remplaçant les équations (2.7) dans (2.6), donnant

$$D(\hat{X}) = D + \frac{1}{2} \frac{\partial D^T}{\partial X} \hat{X} \tag{2.8}$$

Pour les points à faible contraste, la valeur absolue de la fonction est petite, alors on applique tout simplement un seuil pour les supprimer.

Enfin, une analyse des rapports des valeurs propres de la matrice carrée hessienne permet de supprimer les points d'intérêt relevés sur les contours à faible courbure. Pour rappel, la matrice hessienne d'une fonction numérique  $f$  est la matrice carrée, comme indiqué  $h(f)$ , de ses secondes dérivée partielle du gaussien. Cette matrice est définie par :

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \quad (2.9)$$

Il est facile de filtrer les points détectés sur un contour en analysant la trace et le déterminant de la matrice hessienne, on peut l'écrire :

$$\frac{Tr(H)^2}{Det(H)} = \frac{(\alpha+\beta)^2}{\alpha\beta} = \frac{(r\beta+\beta)^2}{r\beta^2} = \frac{(r+1)^2}{r} \quad (2.10)$$

Ou :  $\alpha = r\beta$

La quantité  $(r+1)^2/2$  est minimale lorsque les deux valeurs propres sont égale et elle augmente avec  $r$ . [16]

### 3.3. Assignment d'orientation

En attribuant une orientation cohérente à chaque point clé en fonction des propriétés de l'image locale, le descripteur du point clé peut être représenté par rapport à cette orientation, et ainsi, atteindre l'invariance de rotation de l'image. Cette approche contraste avec les descripteurs invariants d'orientation de Schmid et Mohr (1997), où, chaque propriété d'image s'appuie sur une mesure invariante en rotation. L'inconvénient de cette approche est qu'elle limite les descripteurs qui peuvent être utilisés et élimine les informations d'image en n'exigeant pas que toutes les mesures soient basées sur une rotation constante.

Pour un point-clé donné  $(x, y, \sigma)$ . Le calcul se fait sur  $L$  qui est le gradient de la pyramide dont le paramètre est plus proche de facteur de l'échelle du point, de sorte que tous les calculs sont effectués de la façon invariante à l'échelle.

Pour chaque position dans un voisinage du point-clé, on estime le gradient par différences finies symétriques, puis son amplitude  $m(x, y)$ , et l'orientation,  $\theta(x, y)$ , dont leurs formules sont données par les formules suivantes. [18]

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2} \quad (2.11)$$

$$\theta(x, y) = \tan^{-1} ((L(x, y+1) - L(x, y-1)) / (L(x+1, y) - L(x-1, y))) \quad (2.12)$$

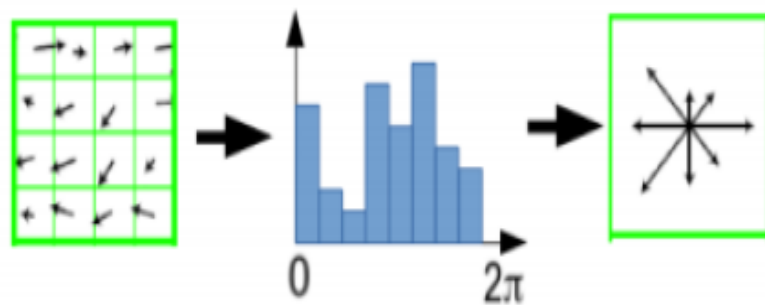


L'histogramme d'orientation comporte 36 intervalles couvrant chacun 10 degrés d'angle. Chaque point ajouté à l'histogramme est pondéré par l'amplitude du gradient, et aussi par une fenêtre circulaire gaussienne avec un écart type qui est 1,5 fois l'échelle du point clé.

Les pics de l'histogramme d'orientation correspondent aux orientations dominantes des gradients locaux (Voir figure 2.3).

Toutes les orientations dominantes qui permettent d'atteindre au moins 80% de la valeur maximale sont prises en compte, ce qui nécessite, la création des points clés supplémentaires qui ne varient que selon leurs orientations principales.

À la fin de cette étape, un point clé est déterminé à quatre paramètres  $(x, y, \sigma, \theta)$ . Il est à noter qu'il est tout à fait possible qu'il y ait plusieurs points clés sur la même image qui ne diffèrent que par l'un de ces quatre paramètres (par exemple, facteur d'échelle ou orientation). [20]



**Figure 2.3:** construction de l'histogramme des orientations.[20]

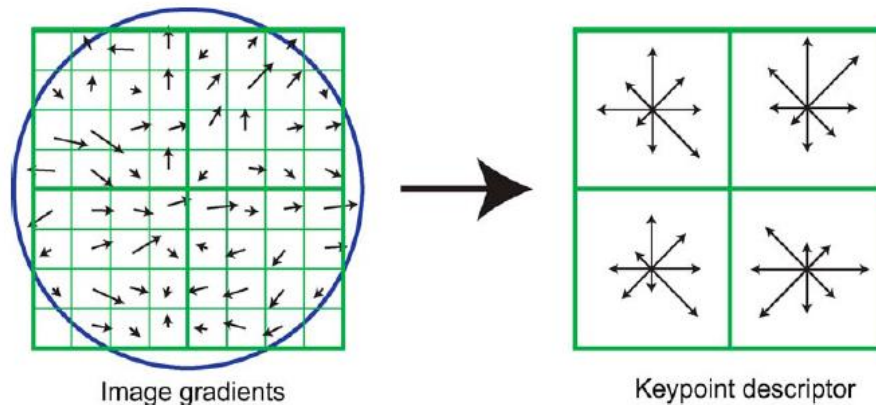
### 3.4. Descripteur de point-clé

Une fois que les points clés qui sont liés aux facteurs et aux orientations d'échelle sont détectés ; l'étape suivante est le calcul des vecteurs descripteurs, traduisant numériquement chacun de ces points clés. À ce moment-là, des traitements supplémentaires assureront un pouvoir discriminant supplémentaire en rendant les descripteurs invariants sur d'autres transformations telles que la luminosité, le changement de point de vue 3D...etc. Cette étape est effectuée sur l'image lissée avec le paramètre de facteur d'échelle le plus proche du point clé considéré.

Sur ce point, nous commençons à modifier le système de coordonnées locales pour assurer l'invariance à la rotation, en utilisant un angle de rotation égal à l'orientation du point clé, ensuite on considère toujours autour du point clé, une région de 16 x16 pixels, divisée en 4x4 zones de 4x4

pixels chacune. Pour chaque zone calculée, un histogramme des orientations comportant 8 intervalles.

A chaque point de la zone, l'orientation et l'amplitude du gradient sont calculées comme précédemment. L'orientation précise la période qui doit incrémenter dans l'histogramme, Ce qui nécessite, d'une part, une pondération avec une amplitude, et d'autre part, une fenêtre gaussienne centrée sur le point clé, de paramètre égale à 1.5 fois le facteur d'échelle du point clé. [20]



**Figure 2.4:** Construction d'un descripteur SIFT. [18]

Par la suite, 16 histogrammes sont concaténés et normalisés à 8 intervalles dans chacun. Afin de réduire la sensibilité des descripteurs aux changements de luminosité, les valeurs sont plafonnées à 0,2 et l'histogramme est à nouveau normalisé le résultat est un vecteur descripteur de  $4 \times 4 \times 8 = 128$  éléments pour chaque point clé. [20]

#### 4. Histogramme du gradient orienté (HOG)

De nouvelles méthodes d'extractions de caractéristiques ont été explorées dans le domaine de vision par ordinateur et en traitement d'images, en particulier la méthode de l'histogramme du gradient orienté « HOG ».

L'histogramme des gradients orientés « HOG » est un descripteur de caractéristiques exploité dans plusieurs branches, et surtout dans l'imagerie, ses concepts ont été décrits par Robert K. McCannell de Wayland Research Inc, sans utiliser le terme « HOG » dans une demande de brevet en 1986. [21]

En 1994, ses concepts ont été utilisés par Mitsubishi Electric Research Laboratories.

Cependant, l'exploitation ne s'est généralisée qu'en 2005 lorsque Navneet Dalal et Bill Triggs, chercheurs de l'Institut national français de recherche en informatique et en automatisation (INRIA), ont présenté leurs travaux sur les descripteurs « HOG » à la Conférence sur la vision par ordinateur et les modèles.

L'idée clé derrière les descripteurs « HOG » est le pouvoir de décrire l'apparence et la forme de l'objet local dans une image par la distribution des gradients d'intensités ou des directions des bords. La mise en œuvre de ces descripteurs peut être réalisée en divisant l'image (le visage dans l'image) en une grille de régions connectées appelées cellules. Ensuite un histogramme de gradient à neuf bandes se forme après l'analyse des gradients des pixels de chacune de ces cellules. La concaténation de ces histogrammes représente alors le descripteur.

## **4. 1. Les étapes d'implémentation du descripteur HOG**

### **4. 1.1. Normalisation de l'image de luminance**

Notre système reçoit en entrée des images convertis en niveau de gris, ensuite normalisées selon l'information de luminance. Cela s'appelle l'étape de prétraitement en fait la normalisation du descripteur « HOG » lui-même s'avèrent suffisante.

### **4. 1.2 .Calcul de l'image gradient**

Pour former le descripteur ; on doit passer par une étape très critique, c'est le calcul du gradient. On dispose de plusieurs manières pour le calculer, cependant, cette variation de méthodes de calcul influe sur les performances du descripteur. Il s'agit de calculer le gradient pour tous les points de l'image, généralement deux filtres dérivatifs sont appliqués sur l'image, le premier est horizontal, le deuxième est vertical, et peuvent être des masques de dérivation simple 1-D (centré et non centré), des opérateurs 2-D de Sobel, ou encore des opérateurs récursifs de Deriche.

Une opération de lissage gaussien 2-D est souhaitable afin de réduire le bruit et de limiter le nombre de contours. Différents filtres dérivatifs ont été testés par Dalal et Triggs:

- Filtre dérivatif 1-D non centré
- Filtre dérivatif 1-D centré

Notant que pour les images couleurs, le gradient se calcule séparément pour chaque composante, et on retient pour chaque pixel le gradient de plus grande norme.

### 1.3. Calcul de l’histogramme

Les histogrammes de l'orientation des gradients se créent sur des cellules carrées de petites tailles (4x4 à 12x12 pixels), chaque pixel de bloc (cellule) vote alors pour une classe de l'histogramme uniforme en fonction de l'orientation du gradient à ce point, que ce soit de 0 à 180° (cas non signé) ou de 0 à 360° (cas signé). Le vote du pixel est pondéré par l'intensité du gradient en ce point.

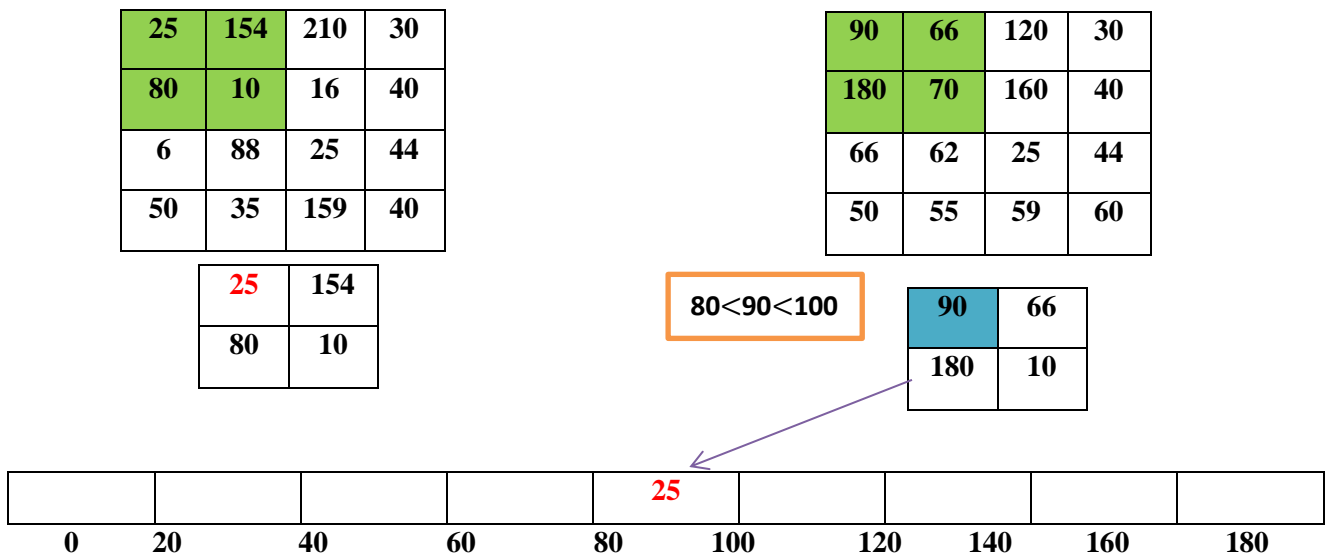


Figure 2.5: calcul de l’histogramme de gradient.

#### 4.1.4. Formation et normalisation des blocs

Les forces de gradient varient sur une large plage et cela est dû aux variations d’illumination et de contraste, ce qui cause une certaine disparité. Pour les éviter, une étape de normalisation des descripteurs s'avère très efficace, elle permet ainsi l’introduction de la redondance dans le descripteur. De ce fait les auteurs rassemblent plusieurs cellules dans un bloc, qui est l'unité sur laquelle est effectuée la normalisation. Les blocs se recouvrent, donc une même cellule participe autant de fois au descripteur final. Ensuite Le descripteur de chaque bloc est normalisé d’une manière indépendante afin d’avoir une norme constante par la division de chaque classe d'histogramme dans le bloc.

La répétition de cette opération sur tous les blocs adjacents permet d’obtenir le descripteur HOG final, la taille des cellules doit être en compromis avec la taille des blocs. En effet, à chaque fois que le nombre de de cellules dans un bloc augmente, le descripteur est plus discriminant. [22]

<b>B1</b>	<b>B2</b>	<b>B3</b>	<b>B4</b>
<b>B5</b>	<b>B6</b>	<b>B7</b>	<b>B8</b>
<b>B9</b>	<b>B10</b>	<b>B11</b>	<b>B12</b>
<b>B13</b>	<b>B14</b>	<b>B15</b>	<b>B16</b>

<b>B1</b>	<b>B2</b>	<b>B5</b>	<b>B6</b>	<b>B2</b>	<b>B3</b>	<b>B6</b>	<b>B7</b>
-----------	-----------	-----------	-----------	-----------	-----------	-----------	-----------

**Figure 2.6:** Normalisation des blocs.

## 5. Classification

La classification automatique est l'analyse qui permet d'organiser et de classer les caractéristiques d'une image afin de lui attribuer une classe, à l'aide d'un système de classification. On possède deux principales catégories de classification dépendantes des informations disponibles sur les données à classer : classification supervisée et classification non supervisée (clustering).

Dans la première approche, chaque photo est associée à une étiquette qui décrit sa classe d'appartenance et cela se fait à partir d'une base de données dite d'apprentissage, tandis que dans la seconde approche les données sont classées sans avoir besoin de cette base de données. Dans les études scientifiques, plusieurs méthodes de classification ont été présentées dont les classifieurs : KNN (K plus proches voisins), les réseaux de neurones, SVM (support vecteur machine) et beaucoup d'autres font partie.

### 5.1. Classifieur KNN

L'algorithme « KNN » ou « K-Nearest neighbors » est l'un des algorithmes d'apprentissage les plus faciles à mettre en œuvre, qui appartient à la classe « supervisée ». C'est-à-dire qu'il doit essentiellement posséder un ensemble d'apprentissages auquel se référer durant son classement. [23]

D'une manière générale l'algorithme d'apprentissage reçoit un ensemble de données qui est déjà étiqueté avec des valeurs de sorties correspondantes, selon ces données il lance son entraînement, puis il définit un modèle de prédiction.

Dans le cas de « KNN » l'idée est comme suit : il cherche les k plus proches voisins du nouveau cas, puis, il choisit parmi les candidats trouvés le résultat le plus proche et le plus fréquent. Ensuite, l'algorithme assigne l'étiquette de cette donnée à la nouvelle observation « inconnue ».

A savoir que le  $k$  dans la formule "k plus proches voisins" veut dire qu'on peut prendre en compte un nombre fixe  $K$  de voisins au lieu de se contenter du seul voisin le plus proche de l'observation inconnue.

A la fin de ces procédures, et en se basant sur la classe majoritaire, nous pouvons faire une prédiction.

## 5.2. Classifieur SVM :

"Support Vector Machine" ou "Séparateur à Vaste Marge" est un classifieur discriminant paramétrique qui permet d'établir un hyperplan séparateur de distance(marge) maximale entre les différents groupes de données représentant chacune des classes, d'où l'appellation "séparateur de marge large". Cependant, Il existe une infinité de droites séparatrices des deux classes mais uniquement une qui assure une distance maximale aux exemples aux frontières des classes. Ces exemples de frontières sont appelés vecteurs de support. [24]

Pour trouver cette frontière séparatrice, nous devons d'abord posséder une série de données d'entraînement « SVM » Nous fournissons à l'algorithme un ensemble de données ayant déjà leurs classes connues. Le « SVM » à son tour va déterminer la frontière la plus admissible. Après la phase d'entraînement, le « SVM » a appris où est l'emplacement de la frontière à partir des données d'entraînement, c'est-à-dire leurs caractéristiques seront utilisées pour prédire le groupe auquel un nouvel enregistrement doit appartenir.

## 6. Conclusion :

Dans ce chapitre nous avons présenté brièvement quelques techniques d'extractions de caractéristiques utilisées dans la vision artificielle (apprentissage supervisé) à savoir, la méthode LBP, SIFT et le descripteur HOG, en expliquant leurs principes de fonctionnement et leurs étapes d'implémentations.

Ces méthodes consistent à représenter l'ensemble des données initial à savoir les images d'entrée par un nouvel ensemble plus réduit, construit à partir du nouvel ensemble de caractéristiques obtenues après la transformation en utilisant les techniques citées ci-dessous.

De plus, on a pu fournir un aperçu un peu explicatif de la classification et ses principales techniques.

Le chapitre suivant, illustre les détails de la conception, ainsi que les méthodes choisies pour la réalisation de notre application.

# Chapitre 3

Application pour la détection  
de visage et port du masque

## **1. Introduction :**

Ce dernier chapitre est réservé pour l'implémentation de notre application qui permet la détection du visage et nous dire si le masque est porté ou pas. Nous avons utilisé et comparé deux descripteurs pour extraire les caractéristiques du visage détecté, la méthode de HOG et la méthode LBP. En premier lieu nous allons présenter le langage et l'environnement de programmation choisis ainsi que le matériel utilisé sans oublier la base de données utilisée. En second lieu nous exposerons notre application et les résultats obtenus et commentés à la fin.

## **2. Présentation d'outils de développement**

### **2.1. Logiciel et langage d'implémentation**

Le logiciel pour la réalisation de notre système est le MATLAB (MATrixLABoratory), qu'est un logiciel interactif basé sur le calcul matriciel à syntaxe simple. Connu pour sa facilité d'apprentissage et d'implémentation dans de divers problèmes scientifiques, ainsi que pour sa capacité de résoudre de différents problèmes numériques complexes en peu de temps requis.

### **2.2. Environnement de travail**

- Un ordinateur portable HP DESKTOP-FMIUDJV
- Windows 10 Professionnel
- Processeur Intel(R) Core (TM) i5-5200U CPU @ 2.20GHz 2.20 GHz
- Mémoire installée (RAM) 4,00Go.
- Type du système : Système d'exploitation 64 bits, processeur x64.

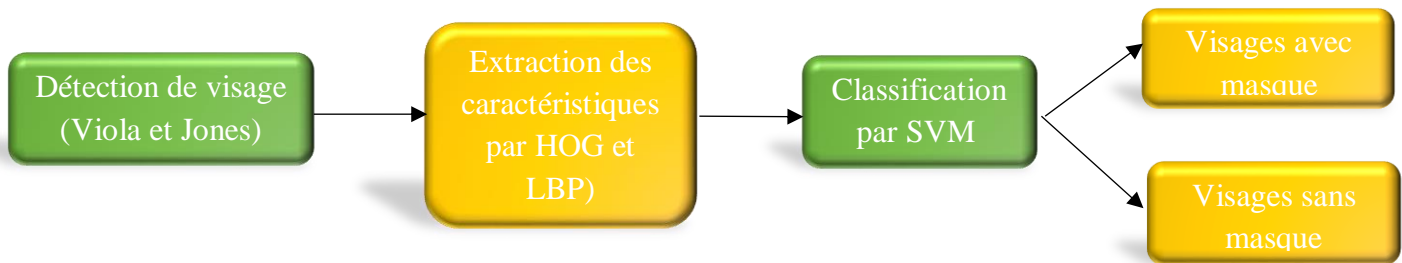
## **3. Présentation de l'application**

Notre travail a pour but de réaliser la détection automatique d'un seul ou plusieurs visages avec une confirmation pour le port du masque.

La détection des visages est appliquée en utilisant l'algorithme de Viola et Jones sur des images, il utilise les différents filtres de Haar pour détecter les yeux, le nez et la bouche. L'étape extraction des caractéristiques permettant de décrire les informations pertinentes contenues dans l'image, pour cela nous avons utilisé deux descripteurs différents à savoir « HOG » et « LBP » ce qui nous amène à la fin à faire une comparaison entre les résultats correspondants.



Pour classer nos images et prendre une décision du port du masque ou non au niveau du visage, nous avons utilisé le classifieur « SVM ».



**Figure 3.1:** Diagramme de notre application.

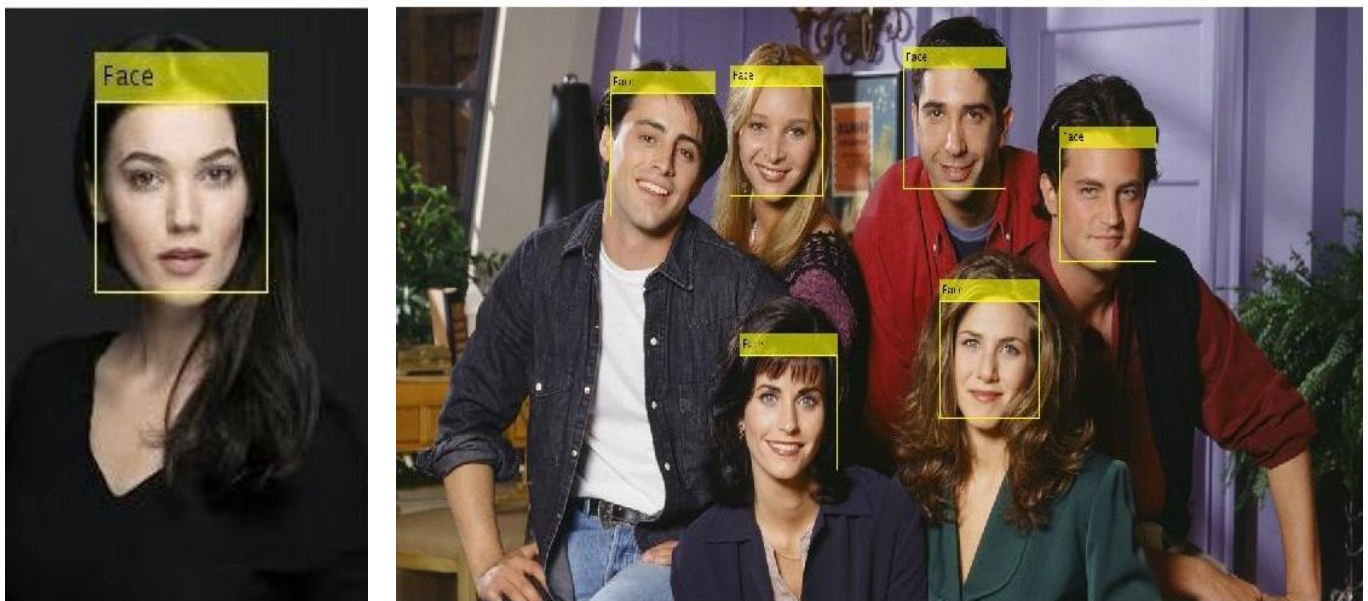
## 4. tests et résultats

### 4.1. Détection de visage

Cette étape est la première procédure dans notre application, on l'a réalisé en utilisant l'algorithme de Viola et Jones.

#### 4.1.1. Test sur des images positives

##### ➤ Visage en face



**Figure 3.2 :** Détection d'un ou plusieurs visages.

## 4.1.2. Les différents cas de dysfonctionnement de l'algorithme

- Pour les visages penchés et inclinés, l'algorithme de Viola et Jones ne fonctionne pas car il est adapté pour les visages regardant directement vers l'appareil photo.



**Figure 3.3:** Problème d'inclinaison des visages lors de la détection.

- Les visages Occlus et partiellement cachés, l'algorithme n'arrive pas à les détecter.



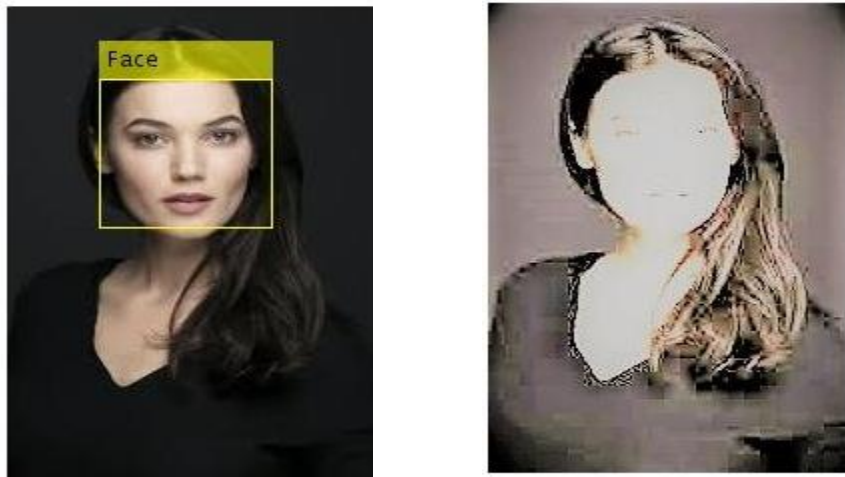
**Figure 3.4:** Résultat de la détection des visages partiellement cachés.

- Les expressions faciales peuvent modifier la géométrie d'un visage, ce qui influe sur la détection



**Figure 3.5:** Résultat dans le cas d'un visage avec des expressions faciales.

- L'illumination et les changements de l'éclairage provoquent l'apparition des ombres accentuant ou masquant quelques caractéristiques du visage engendrant la non détection du visage.



**Figure 3.6:** Résultat de la détection avec différents éclairages.

- Les faux positifs ou visages ou ces cas apparaissent généralement dans les régions ayant la même couleur que celle de la peau du visage ou si une forme d'un objet quelconque et le visage contient une ressemblance significative



Figure 3.7: Image montrant la détection des faux positifs.

## 4.2. Extraction des caractéristiques

Cette étape représente le cœur de notre système de détection de visage. Pour la réalisation de notre projet on a choisi deux descripteurs d'extraction de caractéristique du visage à savoir :

### 4.2.1. Extraction des caractéristiques par « HOG »

Après avoir détecté le visage nous pouvons effectuer l'extraction de ses caractéristiques pour caractériser le modèle du visage en utilisant la fonction le descripteur **HOG**. Cette étape tend à représenter une image ou un patch d'image en les simplifiant par l'extraction des informations utiles et en jetant des informations secondaire non utiles afin de former un vecteur ou matrice de caractéristiques. Cette dernière sera introduite dans un algorithme de classification d'images et comparés à une base de données pour trouver la meilleure correspondance et optimiser la décision finale.

L'histogramme de gradients est calculé pour chaque 32x32 cellule :

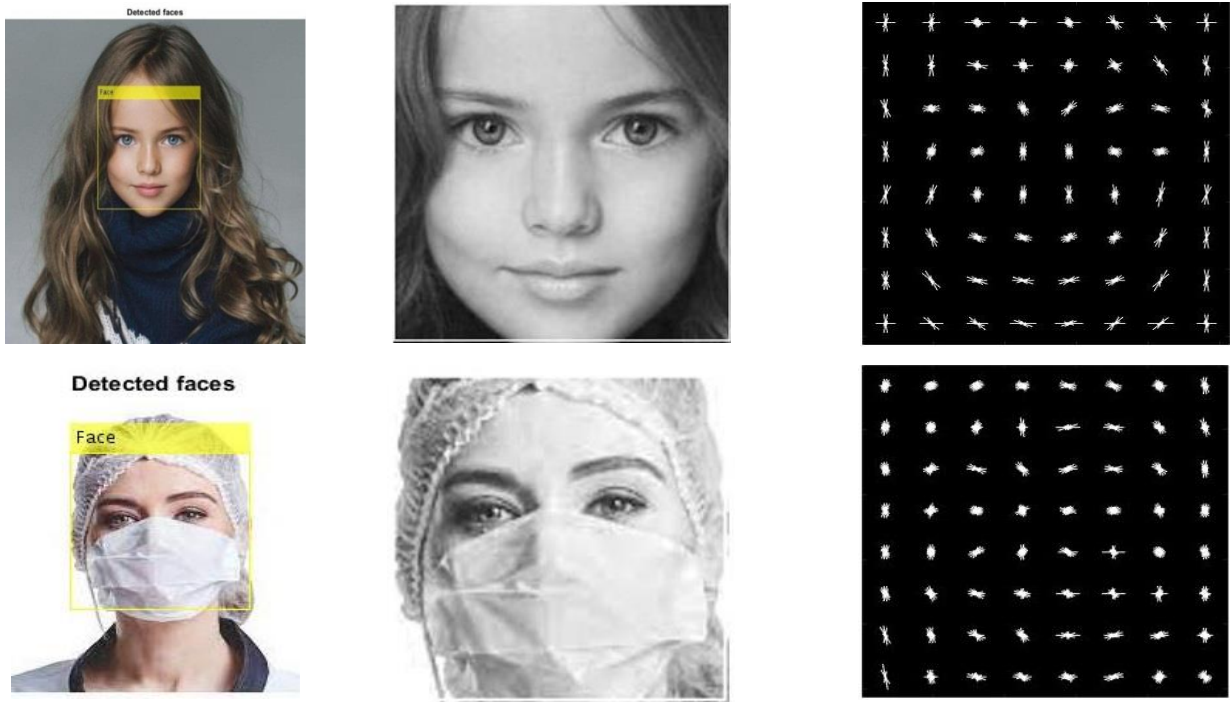


Figure 3.8: Détection et extraction des caractéristiques d'un visage avec et sans masque avec HOG.

#### 4.2.2. Extraction des caractéristiques par « LBP »

Le descripteur textural LBP permet d'analyser les images en un temps-réel, de plus il n'est pas influencé par les changements mono toniques de l'illumination, c'est-à-dire un changement global de la luminosité de l'image n'implique aucune variation dans l'image LBP.

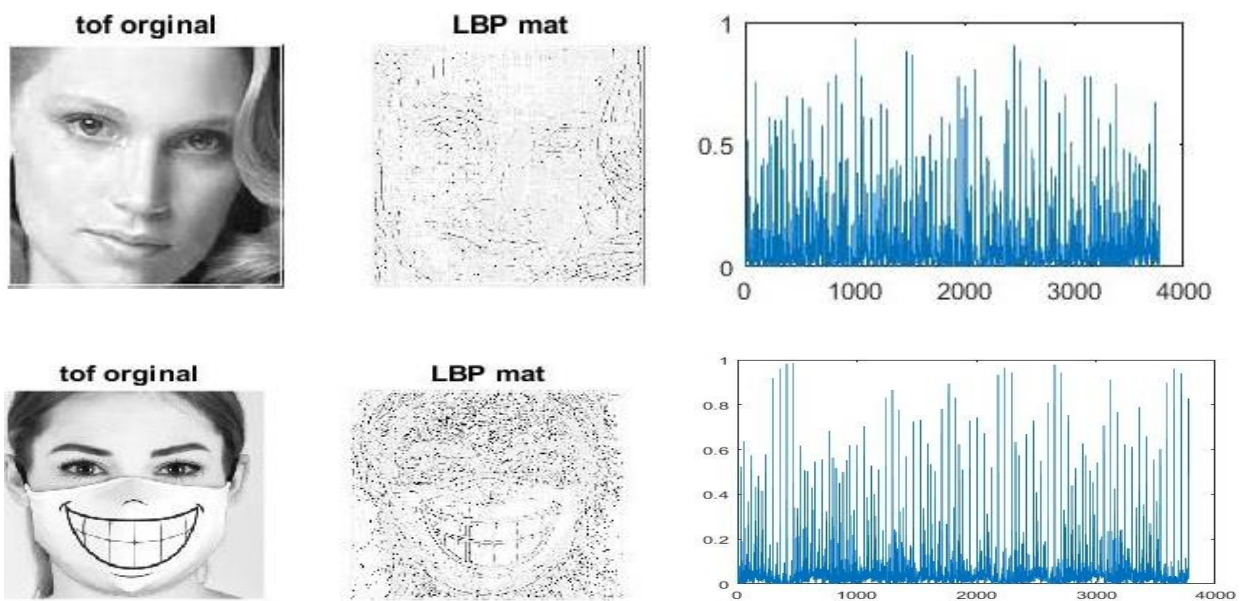


Figure 3.9: Extraction des caractéristiques avec et sans masque avec le descripteur LBP.

## 5. Classification

C'est la dernière étape de notre système de détection, elle vient Juste Après l'extraction du vecteur caractéristique. Elle consiste à attribuer au visage sa classe c'est-à-dire la personne qui lui ressemble le plus dans l'ensemble d'apprentissage. Dans notre travail nous allons utiliser le classifieur « SVM » (support vecteur machine) appartenant à la catégorie des méthodes de classification supervisée.

### 5.1. Création de bases de données pour l'entraînement

Pour lancer notre entraînement et notre opération d'apprentissage, nous avons créé deux bases de données comportant environ de 60 images de visages pour chacune, l'une est avec masque, l'autre est sans masque, téléchargées par internet. Nous avons appliqué l'extraction des caractéristiques des visages de ces deux bases avec les deux descripteurs « HOG » et « LBP » différemment puis nous avons enregistré les vecteurs sous forme de matrices, c'est-à-dire deux matrices pour chaque base seront enregistrées.

Pour lancer nos tests, nous avons utilisé une base de données test, constituée d'un nombre d'images, de 78 avec masque et 62 sans masque.



Figure 3.10: Quelques images de la base de données des visages sans masque.



Figure 3.11: Quelques images de la base de données des visages avec masque.

## 6. Résultats et discussions

Dans cette étape nous testons différentes images de notre bases de données (visages avec masques dites 'positifs' et d'autres sans masques dites 'negatifs').

Nous prenons les resultats en utilisant chaque descripteur et cela afin de calculer les performances de chacun.



**Figure 3.12:** Test du port ou non port du masque sur un seul visage.



**Figure 3.13 :** Test sur un visage portant un masque imprimé.



**Figure 3.14 :** Test sur un visage portant un masque transparent.



**Figure 3.15:** Test du port ou non port du masque sur plusieurs visages.

Les résultats sont évalués en calculant deux mesures à savoir [25] :

- (TPR) : c'est le Taux de vrais positifs, c'est-à-dire la personne portant le masque est correctement identifiée comme portant le masque, elle est calculée par :

$$TPR = \frac{TP}{TP + FN} \tag{3.1}$$

- (FPR) : Taux de faux positifs, c'est-à-dire la personne ne portant pas de masque est identifiée à tort comme étant avec masque, elle est calculée par :

$$FPR = \frac{FP}{FP + TN} \tag{3.2}$$

- (ACC) : Précision, c'est-à-dire le rapport entre le nombre total de prédictions correctes de la classe et la classe réelle de tout l'ensemble de données, elle est calculée par :

$$ACC = \frac{TP + TN}{TP + FN + FP + TN} \tag{3.3}$$



Notons que TP est le Vrai Positif (le port du masque est prédit correctement), TN est Vrai Négatif (sans masque est prédit correctement), FP est Faux Positif (sans masque est prédit comme avec masque à tort) et FN est Faux Négatif (avec masque prédit comme sans masque à tort).

HOG [16 16] Nbre de caracteristiques: 8100	HOG [24 24] Nbre de caracteristiques: 2916	HOG [32 32] Nbre de caracteristiques: 1764	HOG [64 64] Nbre de caracteristiques: 324
(Masque) : 78 Err :00 (FN) Bon :78 (TP)	(Masque) : 78 Err :00 (FN) Bon :78 (TP)	(Masque) : 78 Err :00 (FN) Bon :78 (TP)	(Masque) : 78 Err :00 (FN) Bon :78 (TP)
(sans Masque) : 62 Err :00 (FP) Bon :62 (TN)	(sans Masque) : 62 Err :00 (FP) Bon :62 (TN)	(sans Masque) : 62 Err :00 (FP) Bon :62 (TN)	(sans Masque) : 62 Err :00 (FP) Bon :62 (TN)
(ORL) : 340 Err : 02 (FP) Bon :338 (TN)	(ORL) : 340 Err : 01 (FP) Bon :339 (TN)	(ORL) : 340 Err : 03 (FP) Bon :337 (TN)	(ORL) : 340 Err : 15 (FP) Bon :325 (TN)
<b>TPR:</b> 100% <b>FPR:</b> 0.50% <b>ACC:</b> 99.58%	<b>TPR:</b> 100% <b>FPR:</b> 0.25% <b>ACC:</b> 99.79%	<b>TPR:</b> 100% <b>FPR:</b> 0.75% <b>ACC:</b> 99.37%	<b>TPR:</b> 100% <b>FPR:</b> 3.73% <b>ACC:</b> 96.88%

**Tableau 3.1:** les performances de classification pour les deux bases de données avec le descripteur « HOG ».

LBP [16 16] Nbre de caracteristiques: 15104	LBP [24 24] Nbre de caracteristiques: 5900	LBP [32 32] Nbre de caracteristiques: 3776	LBP [64 64] Nbre de caracteristiques: 944
(Masque) : 78 Err :02 (FN) Bon :76 (TP)	(Masque) : 78 Err :01 (FN) Bon :77 (TP)	(Masque) : 78 Err :01 (FN) Bon :77 (TP)	(Masque) : 78 Err :00 (FN) Bon :78 (TP)
(sans Masque) : 62 Err :00 (FP) Bon :62 (TN)	(sans Masque) : 62 Err :00 (FP) Bon :62 (TN)	(sans Masque) : 62 Err :00 (FP) Bon :62 (TN)	(sans Masque) : 62 Err :01 (FP) Bon :61 (TN)
(ORL) : 340 Err : 00 (FP) Bon :340 (TN)	(ORL) : 340 Err : 00 (FP) Bon :340 (TN)	(ORL) : 340 Err : 00 (FP) Bon :340 (TN)	(ORL) : 340 Err : 00 (FP) Bon :340 (TN)
<b>TPR:</b> 97.44% <b>FPR:</b> 00% <b>ACC:</b> 99.58%	<b>TPR:</b> 98.72% <b>FPR:</b> 00% <b>ACC:</b> 99.79%	<b>TPR:</b> 98.72% <b>FPR:</b> 00% <b>ACC:</b> 99.79%	<b>TPR:</b> 100% <b>FPR:</b> 0.25% <b>ACC:</b> 99.79%

**Tableau 3.2:** les performances de classification pour les deux bases de données avec le descripteur « LBP ».

D'après le tableau (3.1) le descripteur HOG avec la taille des blocs [24 24 ] donne des meilleures performances à savoir, les taux des vrais positifs est de 100% avec un plus petit taux des faux positifs.

Le tableau (3.2) illustre que le descripteur LBP avec la taille des blocs [32 32 ] donne des meilleures performances à savoir, les taux des vrais positifs est de 98.72 % avec un nul au niveau du taux des faux positifs avec un minimum de temps de calcul.

En comparant les deux descripteurs et en tenant compte que notre application doit signaler au maximum les personnes ayant un masque tout en minimisant les intrus ( les personnes qui n'ont pas de masque ). Le descripteur LBP répond à notre but.

## **7. Conclusion**

Dans ce chapitre nous avons présenté notre application qui commence par la détection du visage en utilisant l'algorithme de Viola et Jones et cela pour extraire les caractéristiques. Cette opération est appliquée en utilisant deux descripteurs différents à savoir : « LBP » et « HOG ». le classifieur utilisé est le « SVM ». nous avons utilisé une base de données qui est partagée en deux parties la première pour l'apprentissage et l'entraînement , la seconde est pour tester notre application.

Les résultats de notre classification sont satisfaisantes, montrent clairement que le descripteur LBP avec la taille des blocs [32 32] donne des meilleurs résultats avec un taux élevé des TPR et un faible taux de FPR avec un minimum de temps de calcul.

# Conclusion Générale

### CONCLUSION GENERALE

Pour faire face à la pandémie du COVID-19, L'élaboration d'un système de détection automatique du port de masque contribue à réduire la propagation du virus, c'est dans ce contexte que s'inscrit le travail qu'on a développé dans notre mémoire.

En premier nous avons donné un aperçu général sur les méthodes de détection de visage Pour en choisir au final l'algorithme de Viola et Jones pour la réalisation de notre système. De plus on a abordé une variété de techniques d'extraction de caractéristiques telles que SIFT, HOG et LBP, mais on n'a opté que pour les deux derniers. Par ailleurs, on a expliqué brièvement la dernière phase de notre système de détection qu'est la classification en donnant deux exemples de classifieurs : KNN et SVM.

Pour la réalisation de notre application nous avons exploité la méthode de viola et Jones pour détecter un ou plusieurs visages et les extraire (régions d'intérêt), en suite on a utilisé les deux descripteurs HOG et LBP pour extraire les caractéristiques. Par la suite nous avons créé une base de données contenant des images de visages avec et sans masques pour lancer l'entraînement, tout en utilisant indépendamment les deux descripteurs mentionnés en haut. Une fois que les caractéristiques extraites sont enregistrées dans des matrices, nous avons entamé la phase de classification, où on a choisi le SVM comme classifieur pour faire notre test, D'ailleurs nous avons pris l'exemple sur un visage ou plusieurs visages pour distinguer ceux portant le masque de ceux qui ne le portent pas.

Au final de notre travail on a lancé plusieurs tests différents avec de différentes bases de données y compris la base ORL et avec de différentes tailles de cellules, pour comparer les performances de chacun des deux descripteurs « HOG » et « LBP ». Nous avons conclu à la fin que le descripteur « LBP » est le choix idéal pour la détection des visages avec masque.

Comme perspectives et travaux futurs :

- Améliorer le module de détection automatique de visages de sorte qu'il réponde aux différents cas de dysfonctionnement.
- L'utilisation d'une grande base de données dans la phase d'entraînement afin d'améliorer les performances de la classification.
- Utilisation des classifieurs évolutifs pour la détection à savoir, les CNN.

# Bibliographie

***Bibliographie***

- [1]: S. Meivel, K. Indira Devi, S. Uma Maheswari, J. Vijaya Menaka. " *Real time data analysis of face mask detection and social distance measurement using Matlab*". Department of ECE, M.Kumarasamy College of Engineering, India, Department of ICE, GRG Polytechnic College, India. Article, December 2020.
- [2]: M.KHAMMARI. Détection et suivi de visages en temps Réel sur Flux Vidéo. Thèse Présentée en vue de l'obtention du diplôme De doctorat troisième cycle LMD. Université Mokhtar Badji- Annaba ,2016.
- [3]: <http://davincibot.org/inmoov-robot-cognitif/>,consulté le 17.04.2022.
- [4]: Cheng-chinchiang, wen-kai tai, mau-tsuen yang, yi-tinghuang, and chijaunghuang : " à novelmethod for detectinglips, eyes and faces in real time Real-time imaging", 9(4) :277,287, 2003.
- [5]: S.Satoh, Y.Nakamura, and T. Kanade, naming and detecting faces in news videos, *IEEE Multimedia*, 6:22-35, 1999.
- [6]: A. SHARIFARA, M. S. MOHD RAHIM, Y. ANISI, "A general review of human face detection including a study of neural networks and Haar feature-based cascade classifier in face detection", *International Symposium on Biometrics and Security Technologies (ISBAST)*, 2014, pp. 73-78, doi: 10.1109/ISBAST.2014.7013097.
- [7]: Ming-Hsuan Yang, David J. Kriegman, Narendra Ahuja, "Detecting Faces in Images: A Survey",*IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, **VOL. 24**, NO. 1, JANUARY 2002.
- [8]:Wenlong Zheng, Suchendra M. Bhandarkar, "Face detection and haking using a boosted adaptative particle filter ". *Journal of visual communication and Imog Representation*, **20(1)**, pp. 9-27, 2009.
- [9]: P. A. NEGRI, "Détection et Reconnaissance d'objets structurés : Application aux Transports Intelligents", Thèse doctorat, Université Pierre et Marie Curie - Paris VI, France, Septembre 2008.
- [10]: P. VIOLA, M.J. JONES, Robust Real-time Object Detection, *International Journal of Computer Vision* (57), pp. 137–154, 2001.
- [11]: R. LIENHART, J. MAYDT, "An Extended Set of Haar-like Features for Rapid Object Detection ", *Proceedings. International Conference on Image Processing*, 2002, pp. I-I, doi: 10.1109/ICIP.2002.1038171.

- [12] : W. HIZEM, B. DORIZZI, “ *Capteur Intelligent pour la Reconnaissance de Visage* “, Thèse doctorat en Electronique/Informatique, Université Pierre et Marie Curie - Paris 6, 2009.
- [13]: Y. FREUND, R. E. SCHAPIRE, “A decision theoretic generalization of on-line learning and an application to boosting”, *European Conference on Computational Learning Theory*, pp.23, 1995.
- [14] : H. FAOUZI, “ *Apprentissage des réseaux d'ondelette beta base sur la théorie des frames: Application a la détection de visages* “, Mémoire de Magistère, Université de Gabes, 2008.
- [15]: T. OJALA, M. PIETIKAINEN, D. HARWOOD, “A Comparative Study of Texture Measures with Classification Based on Feature Distributions “ : *Pattern Recognition*. **29(1)**, pp. 51-59, 1996.
- [16]: T. AHONEN, A. HADID, M. PIETIKAINEN, “Face description with local binary patterns : Application to face recognition“, *IEEE transactions on pattern analysis and machine intelligence*. **28(12)**, 20372041, 2006.
- [17]: G. L. DAVID, “Object recognition from local scale-invariant features in Computer vision“, *The proceedings of the seventh IEEE international conference on*. **Vol. 2**, pp. 1150–1157, 1999.
- [18]: G. L. DAVID, “Distinctive image features from scale-invariant key points “, *International journal of computer vision*, **60(2)**, pp. 91–110, 2004.
- [19]: M. TREIBER, “ An introduction to object recognition : selected algorithms for a wide variety of application “, *Springer*, **Vol. 202**, pp. 147, 2010.
- [20]: A. BOHI, “ *Descripteurs de Fourier inspirés de la structure du cortex visuel primaire humain Application à la reconnaissance de navires dans le cadre de la surveillance maritime*“, Thèse doctorat en Informatique, Université de Toulon, 2017.
- [21]: Dalal, N., Triggs, B, “Histograms of oriented gradients for human detection, “*IEEE computer society conference on computer vision and pattern recognition*”,vol. 1, pp. 886-893, 2005
- [22]: C. MIGNIOT, *Segmentation de personnes dans les images et les vidéos*, thèse doctorat en Sciences et technologie industrielles, Université de Grenoble, 2012.
- [23]: Saravanan Thirumuruganathan. A detailed introduction to k-nearest neighbor (KNN) algorithm, Article, may 2010.
- [24]: A.DJEFFAL, “Utilisation des méthodes Support Vector Machine (SVM) dans l’analyse des bases de données”, Thèse présentée pour l’obtention du diplôme de Docteur en sciences, université Mohamed Khider, Biskra, Algérie, 2012.

[25] : S.Chelbi, A.Mekhmoukh, A practical implementation of mask detection for COVID-19 using face detection and histogram of oriented gradients. “Australian Journal of Electrical and Electronics Engineering”2021,pp,5,doi.org/10.1080/1448837X.2021.2023071,2021.