



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE IBN KHALDOUN - TIARET

MEMOIRE

Présenté à :

FACULTÉ MATHÉMATIQUES ET INFORMATIQUE
DÉPARTEMENT D'INFORMATIQUE

Pour l'obtention du diplôme de :

MASTER

Spécialité : Génie informatique

Par :

MECHTOUNE Elhacene Lazrag
MASSARIA Brahim Abdeldjebar

Sur le thème

Extraction des caractéristiques discriminantes basées sur LBP pour la classification des textures

Soutenu publiquement le .. / .. /2021 à Tiaret devant le jury composé de :

Mr BEKKI Khadhir	MAA	Université Ibn Khaldoun Tiaret	Président
Mr GAFOUR Yacine	MCB	Université Ibn Khaldoun Tiaret	Encadreur
Mr CHENINE Abdelkader	MAA	Université Ibn Khaldoun Tiaret	Examineur

2020-2021

Remerciement

Nous remercions en premier lieu ALLAH qui nous a éclairé le chemin du savoir et qui nous a donné la volonté et la patience d'achever ce modeste travail de mémoire, notre grand salut sur le premier éducateur notre prophète Mohamed (satisfaction et salut de dieu soit sur lui).

Nous tenons à adresser mes remerciements aux nos parents qui a consenti des sacrifices et prodigué des encouragements tout au long de mes études.

*Nous adressons nos vifs remerciements et nos sincères gratitude, à notre Encadreur Monsieur **GAFOUR Yacine** qui nous a fait l'honneur d'avoir la charge d'encadrer notre travail de mémoire avec grande patience, pour la confiance qu'il a eu en notre projet et surtout pour ses orientations, ainsi que son aide précieuse et ses conseils pour réaliser ce mémoire.*

Nous Remercions également notre jury d'avoir accepté de juger notre travail et nous tenons à remercier également nos collègues pour leur aide de réalisation ce Modeste travail.

Dédicace

A mes très chers parents, source de vie, d'amour et d'affection

A mes chers frères et leurs enfants, source de joie et de bonheur

A toute ma famille, source d'espoir et de motivation

A tous mes amis

A vous cher lecteur

MASSARIA Brahim Abdeldjebar

Dédicace

À mes chers parents, pour tous leurs sacrifices, leur amour, leur tendresse, leur soutien et leurs prières tout au long de mes études,

À mes chères sœurs pour leurs encouragements permanents, et leur soutien moral,

À mes chers frères, pour leur appui et leur encouragement,

À toute ma famille pour leur soutien tout au long de mon parcours universitaire,

Que ce travail soit l'accomplissement de vos vœux tant allégués, et le fruit de votre soutien infailible,

Merci d'être toujours là pour moi.

MECHOUNE Elhacene Lazrag

1 Sommaire

CHAPITRE 1	2
1.1 INTRODUCTION	2
1.2 DEFINITIONS PRELIMINAIRES	2
1.2.1 <i>Notion d'image</i>	2
1.2.1.1 Définition d'une image	2
1.2.1.2 Images en niveaux de gris	3
1.2.1.3 Images en couleur	3
1.2.1.4 Images de dentelle	3
1.2.2 <i>Caractéristiques d'une image</i>	4
1.2.2.1 Pixels	4
1.2.2.2 Dimension	5
1.2.2.3 Résolution	5
1.2.2.4 Bruit	5
1.2.2.5 Histogramme	5
1.2.2.6 Modes colorimétriques	6
1.2.3 <i>Prétraitement d'une image</i>	7
1.2.3.1 Génération des caractéristiques	8
1.2.3.2 Normalisation des données	8
1.2.3.3 Fusion d'information	8
1.3 NOTION DE TEXTURE	8
1.3.1 <i>Définition de texture</i>	9
1.3.1.1 Familles de texture	11
1.4 CONCLUSION	12
CHAPITRE 2	13
2.1 INTRODUCTION	13
2.2 SYSTEMES DE RECONNAISSANCE DE FORME	13
2.2.1 <i>Extraction des caractéristiques</i>	14
2.2.2 <i>Descripteurs de texture</i>	15
2.2.3 <i>Méthodes d'extraction des caractéristiques</i>	16
2.2.3.1 Motif binaire local 'LBP'	17
2.2.3.2 Histogramme de gradient orienté 'HOG'	19
2.3 CLASSIFICATION AUTOMATIQUE	22
2.3.1 <i>Classification supervisée</i>	22
2.3.1.1 Séparateurs à vaste marge 'SVM'	24
2.3.1.2 K-plus proche voisins 'k-NN'	27
2.3.1.3 Réseaux bayésiens naïf	29
2.3.1.4 Réseaux de neurones	30
2.3.1.5 <i>Arbre de décision</i>	33
2.3.1.6 <i>Apprentissage en profondeur</i>	35
2.3.2 <i>Classification non supervisée</i>	36
2.3.2.1 K-means	37
2.4 CONCLUSION	38
CHAPITRE 3	39
3.1 INTRODUCTION	39
3.2 ENVIRONNEMENTS DE TRAVAIL	39
3.2.1 <i>Environnement PyCharm</i>	39
3.2.2 <i>Environnement Python</i>	40
3.2.3 <i>Environnement Qt designer</i>	40

3.3	METHODE PROPOSEE POUR LA CLASSIFICATION DES TEXTURES.....	41
3.3.1	<i>Le descripteur LCV</i>	<i>Error! Bookmark not defined.</i>
3.3.2	<i>Descripteur LBP</i>	41
3.3.3	<i>Descripteur LCV</i>	42
3.3.3.1	<i>Corrélation.....</i>	42
3.3.3.2	<i>Algorithme LCV.....</i>	42
3.4	EXPERIENCE ET RESULTATS	44
3.4.1	<i>Base de données.....</i>	44
3.4.2	<i>Résultats obtenus</i>	45
3.5	CONCLUSION	46

Table de figure

Figure 1 : Palette de 256 couleurs utilisées.	3
Figure 2 : Une image sous forme d'un tableau de pixels.	4
Figure 3 : Résolution d'une image selon le nombre total de pixels.....	5
Figure 4 : Espace de couleur, (a) RVB, (b) CMNJ.....	7
Figure 5 : Schéma d'un système de traitement d'image.....	7
Figure 6 : Vue générale de l'analyse de texture.	11
Figure 7 : Familles de textures : [a-d] Déterministes, [e-h] Observables, [i-l] Stochastiques.....	Error!
Bookmark not defined.	
Figure 8 : Étapes d'extraction des caractéristiques.	13
Figure 9 : Classification des méthodes d'apprentissage automatique.	14
Figure 10 : Illustration de LBP basique.....	18
Figure 11 : Exemples de voisinages avec différentes valeurs de (P, R).	18
Figure 12 : Schéma d'obstruction d'histogramme LBP.	19
Figure 13 : Exemples d'histogramme obtenue dans méthode de HOG.	20
Figure 14 : Expérience basée sur la méthode de HOG.....	21
Figure 15 : Schéma général d'un système d'apprentissage supervisé [44].	23
Figure 16 : Illustration du principe des SVM.....	25
Figure 17: SVM linéaire et non linéaire [53].	26
Figure 18 : Schéma représentant le principe intuitif d'un K-NN [51].....	28
Figure 19 : La structure d'un réseau de neurone CNN [58].	30
Figure 20 : La structure d'un réseau de neurone [61].....	31
Figure 21 : La structure d'un réseau de neurone multicouches MLP [62].	32
Figure 22 : Exemple d'un réseau de convolutions avec deux couches.....	33
Figure 23: La structure générale d'un arbre de décision DT [64].	34
Figure 24 : Schéma général du ML comparé à celui du DL.....	35
Figure 25 : Schéma général d'un système d'apprentissage non supervisé [44].	36
Figure 26: Exemple de classification avec la méthode de k-means.	38
Figure 27 Logo de logiciel PyCharm.....	39
Figure 28: Python logo.....	40
Figure 29 Qt designer logo.	40
Figure 30 Modèle d'apprentissage par la méthode LCV.	42
Figure 31 Architecture d'un système de reconnaissance des textures.....	41
Figure 32 Photo matricielle.	43
Figure 33 Les voisins de la matrice centré,.....	43
Figure 34 Le calcul d'un bit.....	44
Figure 35 : La transformation finale de l'image.	44
Figure 36 Interface de l'application de reconnaissance des expressions textures LCV	45

Résumé

La classification des textures est un problème fondamental dans la vision par ordinateur et le traitement d'images, jouant un rôle important dans un large éventail d'applications qui incluent l'analyse d'images médicales. Le descripteur Local Binary Pattern (LBP) est largement utilisé dans l'analyse de texture en raison de sa simplicité de calcul et de sa robustesse aux changements d'éclairage. Cependant, LBP a des limites pour capturer les informations discriminantes puisque seules les comparaisons de différence d'intensité sont calculées dans une région locale. Pour améliorer les performances de LBP, nous proposons un descripteur s'appelle Local Corrélation Voisinage (LCV) basé sur LBP. Le descripteur LCV est calculé en se basant sur la corrélation autour du pixel centre, et sa dimension est la même de LBP. Les résultats expérimentaux sur la base de données de texture représentatives CURET montrent que le LCV proposé reste à améliorer par rapport à LBP conventionnelle.

Mot clés : LBP, LCV, descripteur, classification de texture

Abstract

Texture classification is a fundamental problem in computer vision and image processing, playing an important role in a wide range of applications which include the analysis of medical images. The Local Binary Pattern (LBP) descriptor is widely used in texture analysis because of its computational simplicity and its robustness to lighting changes. However, LBP has limitations in capturing discriminant information since only intensity difference comparisons are computed in a local region. To improve the performance of LBP, we provide a descriptor called Local Correlation Neighborhood (LCV) based on LBP. The LCV descriptor is calculated based on the correlation around the center pixel, and its dimension is the same as LBP. Experimental results based on CURET representative texture data show that the proposed LCV remains to be improved over conventional LBP.

Keywords: LBP, LCV, descriptor, texture classification

المخلص

يعد تصنيف النسيج مشكلة أساسية في رؤية الكمبيوتر ومعالجة الصور ، ويلعب دورًا مهمًا في مجموعة واسعة من التطبيقات التي تشمل تحليل الصور الطبية. يستخدم واصف النمط الثنائي المحلي (LBP) على نطاق واسع في تحليل النسيج بسبب بساطته الحسابية ومتانته في تغيرات الإضاءة. ومع ذلك ، فإن LBP لديها قيود في التقاط المعلومات التمييزية حيث يتم حساب مقارنات فرق الشدة فقط في منطقة محلية. لتحسين أداء LBP ، نقدم واصفًا يسمى جوار الارتباط المحلي (LCV) استنادًا إلى LBP. يتم حساب واصف LCV بناءً على الارتباط حول بكسل المركز ، ويكون أبعاده هو نفسه LBP تظهر النتائج التجريبية المستندة إلى بيانات النسيج التمثيلية لـ CURET أن LCV المقترح لا يزال بحاجة إلى التحسين على LBP التقليدي.

الكلمات المفتاحية: LBP ، LCV ، الواصف ، تصنيف النسيج

Introduction générale

Depuis plusieurs décennies, les textures du monde réel sont très difficiles à l'analyser, souvent due à la variabilité de la surface, la luminance, des variations d'échelle, et à une inhomogénéité naturelle de la texture. Parmi le problème fondamental existant dans le domaine du traitement des images et de la vision par ordinateur est la classification des textures. Elle peut les trouver dans plusieurs applications comprenant la télédétection, l'analyse d'images médicales, la reconnaissance et la détection d'objets, l'analyse de documents, indexation et la recherches des images par le contenu, etc.

Le but global de la classification de la texture est d'atteindre deux objectifs principaux, le premier est de capturer les informations de texture les plus représentatives et spécifiques d'une classe de texture afin de pouvoir distinguées entre les différentes classes de texture en présence de diverses variations sur l'image (occlusion, rotation, éclairage, point de vue, mise à l'échelle, et bruit). Le deuxième est d'assurer une faible complexité de calcul, pour permettre à la tâche de classification de s'exécuter en temps réel.

Dans ce mémoire, nous proposons une nouvelle méthode pour capturer des informations discriminantes pour représenter les images de texture. Ces informations permettent aux classificateurs de mieux discriminer les classes d'objet les unes des autres parce que les objets appartenant à la même classe présentent une similarité de leurs formes et que ces informations devraient et peuvent être utilisées dans des systèmes de classification des objets.

Le reste du mémoire est organisé comme suit. Dans le chapitre 1, nous présentons une généralité sur les images et les textures. Nous discutons dans le chapitre 2 l'apprentissage automatique et les différents algorithmes utilisés dans la classification. Nous expliquons ensuite notre méthode proposée, la base de données et les résultats expérimentaux dans le chapitre 3. Enfin, nous terminerons ce mémoire avec une conclusion générale.

Chapitre 1

‘Généralités sur les Images et les Textures’

1.1 Introduction

La classification d'images par le contenu consiste à retrouver, dans une base de données, des images visuellement similaires à une image requête. Généralement, le contenu des images est caractérisé par une signature (distribution, vecteur d'attributs...). Dans l'étape de classification, une signature est extraite de l'image requête pour être comparée suivant différentes mesures de similarité à l'ensemble des signatures stockées.

Nous commençons dans ce chapitre par une description générale de l'image et texture et les modèles utilisés pour les extraire et classifier. Ensuite nous décrivons les descripteurs les plus utilisés dans le domaine d'identification du visage et les expressions faciales.

1.2 Définitions préliminaires

1.2.1 Notion d'image

Une image est une représentation graphique d'une scène ou d'un objet situé en général dans un espace tridimensionnel. Son élaboration résulte de la volonté de proposer une entité observable par l'œil humain. Ceci explique d'une part son aspect planaire et d'autre part le fait que l'information élémentaire associée à chaque point de l'image représenté en niveau de gris ou en couleur. Cette information élémentaire est, en général, une grandeur physique liée à la nature du capteur. Une caméra mesure la quantité de lumière émise par l'objet, un récepteur de rayons X mesure la quantité de rayonnement transmise par l'organe (en imagerie médicale) ou la pièce [1]. L'image peut être défini comme une collection d'informations qui se présentait sur un support photographique qui permettait le traitement d'un phénomène bref, une analyse fine des phénomènes enregistrés et bien sûr l'archivage et l'illustration.

1.2.1.1 Définition d'une image

Une image est plutôt difficile à décrire d'une façon générale. Une image est une représentation du monde. En vision par ordinateur, l'image est considérée comme une fonction mathématique de $R \times R$ dans R . Le couplet d'entrée est considéré comme une position spatiale, le singleton de sortie comme l'intensité (couleur ou niveaux de gris) du phénomène physique. Il arrive cependant que l'image soit dite "3D" donc la fonction est de $R \times R \times R$. Les images couleurs peuvent être représentées soit par trois images représentant les trois couleurs fondamentales (rouge, vert, bleu).

L'image numérique est une matrice dont les coordonnées sont divisés en éléments de tailles fixes appelés cellules ou pixels, ayant chacun comme caractéristique un niveau de gris ou de couleurs prélevé à l'emplacement correspondant dans l'image réelle, ou calculé à partir d'une description interne de la scène à représenter [2].

1.2.1.2 Images en niveaux de gris

Le niveau de gris est la valeur de l'intensité lumineuse en un point. La couleur du pixel peut prendre des valeurs allant du noir au blanc en passant par un nombre fini de niveaux intermédiaires. Donc pour représenter les images à niveaux de gris, on peut attribuer à chaque pixel de l'image une valeur correspondant à la quantité de lumière renvoyée. Cette valeur peut être comprise par exemple entre 0 et 255. Chaque pixel n'est donc plus représenté par un bit, mais par un octet. Pour cela, il faut que le matériel utilisé pour afficher l'image soit capable de produire les différents niveaux de gris correspondant. Le nombre de niveaux de gris dépend du nombre de bits utilisés pour décrire la " couleur " de chaque pixel de l'image. Plus ce nombre est important, plus les niveaux possibles sont nombreux [2].

1.2.1.3 Images en couleur

Même s'il est parfois utile de pouvoir représenter des images en noir et blanc, les applications multimédias utilisent le plus souvent des images en couleurs. La représentation des couleurs s'effectue de la même manière que les images monochromes avec cependant quelques particularités. En effet, il faut tout d'abord choisir un modèle de représentation. On peut représenter les couleurs à l'aide de leurs composantes primaires. Les systèmes émettant de la lumière (écrans d'ordinateurs, ...) sont basés sur le principe de la synthèse additive : les couleurs sont composées d'un mélange de rouge, vert et bleu (modèle R.V.B.) [2].

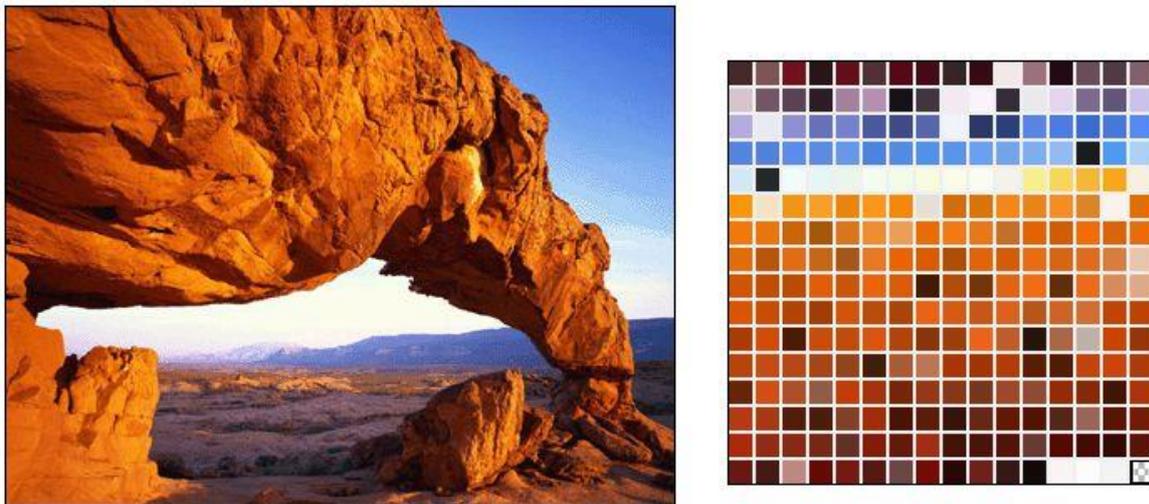


Figure 1 : Palette de 256 couleurs utilisées.

1.2.1.4 Images de dentelle

Les dentelles sont très variées suivant plusieurs critères comme le type des motifs (florale, animale, ...), la nature des motifs (coton, textile, ...), le type des fonds de dentelles et leurs couleurs. Cette diversité des dentelles possède néanmoins une spécificité commune. En effet, il existe un ensemble limité et bien défini de fonds de dentelles.

La reconnaissance des différents types de dentelles en les classant selon leur fond représente un véritable challenge. En effet, outre les conditions d'acquisition qui peuvent être variables, certaines classes de fonds sont très similaires et il est difficile, même pour l'œil humain, de discriminer certaines classes. [2]

1.2.2 Caractéristiques d'une image

L'image est un ensemble structuré d'informations caractérisé par les paramètres suivants :

1.2.2.1 Pixels

Une image numérique est constituée d'un ensemble de points appelés pixels (abréviation de PictureElément) pour former une image. Le pixel représente ainsi le plus petit élément constitutif d'une image numérique. L'ensemble de ces pixels est contenu dans un tableau à deux dimensions constituant l'image :

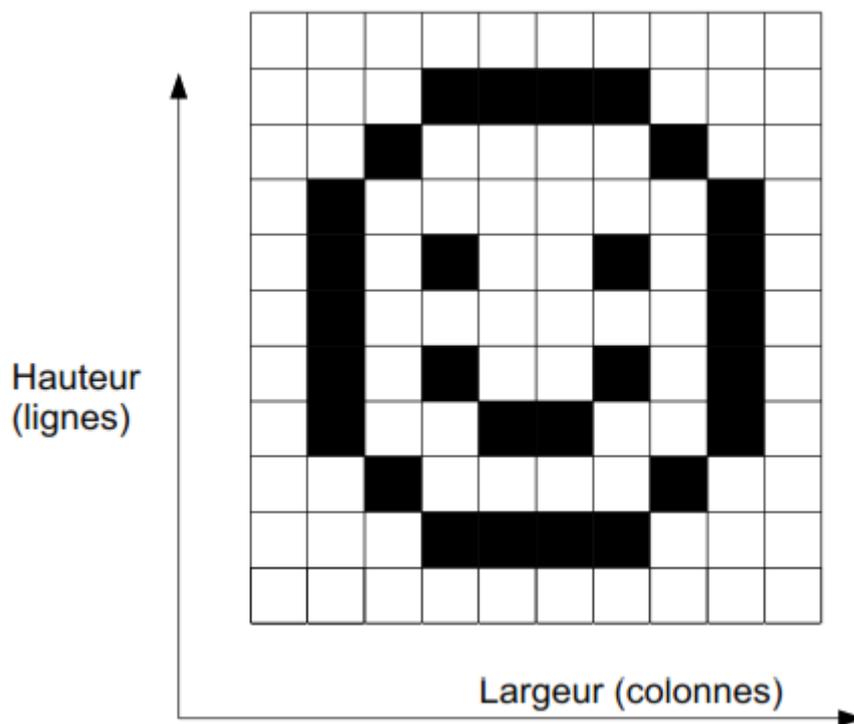


Figure 2 : Une image sous forme d'un tableau de pixels.

Les Principaux formats d'affichage :

Pour afficher ces images, des formats d'affichages standards ont été définis en fonction de l'évolution des capacités matérielles des cartes graphiques et des écrans. Voici les principaux :

- ❖ CGA (320 x 200) 4 couleurs.
- ❖ VGA (640 x 480) 16 couleurs
- ❖ SVGA (800x600) 256 couleurs
- ❖ XGA (1024 x 768) 256 couleurs
- ❖ SXGA (1280 x 1024) en 16 millions de couleurs.

1.2.2.2 Dimension

C'est la taille de l'image. Cette dernière se présente sous forme de matrice dont les éléments sont des valeurs numériques représentatives des intensités lumineuses (pixels). Le nombre de lignes de cette matrice multiplié par le nombre de colonnes nous donne le nombre total de pixels dans une image [2].

1.2.2.3 Résolution

C'est la clarté ou la finesse de détails atteinte par un moniteur ou une imprimante dans la production d'images. Sur les moniteurs d'ordinateurs, la résolution est exprimée en nombre de pixels par unité de mesure (pouce ou centimètre en anglais 'dpi' est dot-per-Inch). On utilise aussi le mot résolution pour désigner le nombre total de pixels affichables horizontalement ou verticalement sur un moniteur plus grand est ce nombre, meilleure est la résolution [2].

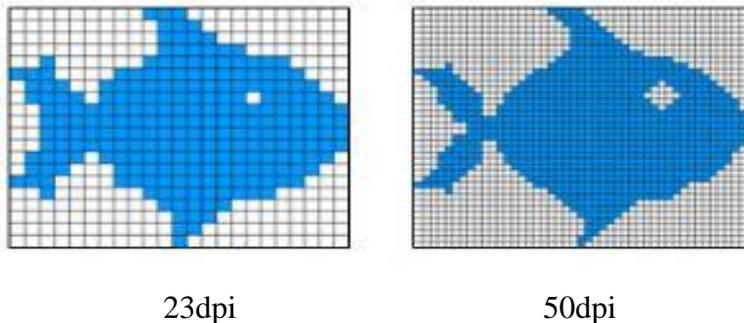


Figure 3 : Résolution d'une image selon le nombre total de pixels.

1.2.2.4 Bruit

Un bruit (parasite) dans une image est considéré comme un phénomène de brusque variation de l'intensité d'un pixel par rapport à ses voisins, il provient de l'éclairage des dispositifs optiques et électroniques du capteur [2].

1.2.2.5 Histogramme

La représentation des descripteurs visuels doit être :

- ❖ **Discriminante** : permettant de différencier entre le contenu des images,
- ❖ **Invariante** : ne variant pas avec la taille ou la luminosité des images,
- ❖ **Compacte** : pour palier au problème de stockage et d'accès.

La représentation la plus usuelle est l'histogramme. Il est invariant aux changements d'orientation, de taille et de position d'un objet dans l'image. Cependant, il ne stocke pas l'information sur les relations spatiales entre les objets. Un histogramme représente le nombre d'apparitions d'un élément dans un ensemble.

Dans le cas où la caractéristique visuelle est la couleur, l'histogramme de descripteurs de couleurs est construit en deux étapes : les couleurs sont quantifiées, puis les couleurs quantifiées sont dénombrées. Chaque composante du vecteur donne la quantité d'une couleur présente dans l'image [3]. Dans le cas d'images complexes, le nombre de couleur quantifié peut être très grand. On doit donc recourir à la quantification pour obtenir une représentation compacte du contenu visuel. Pour ce faire, les histogrammes extraits sont regroupés par classification. On obtient des clusters visuels [4]. Pour rechercher les images les plus similaires à une image requête, il faut comparer les histogrammes pour mesurer la similarité des images correspondantes, ou plus exactement de mesurer leur similarité visuelle.

L'histogramme des niveaux de gris ou des couleurs d'une image est une fonction qui donne la fréquence d'apparition de chaque niveau de gris (ou couleur) dans l'image. Il permet d'extraire un grand nombre d'information sur la distribution des niveaux de gris ou couleur et de voir entre quelles bornes la majorité des niveaux de gris est répartie pour les deux cas d'image (claire ou foncée). Il peut être aussi utilisé pour améliorer la qualité d'une image (rehaussement d'image) en introduisant quelques modifications. Pour diminuer l'erreur de quantification, pour comparer deux images obtenues sous des éclairages différents, ou pour mesurer certaines propriétés sur une image, on modifie souvent l'histogramme correspondant.

1.2.2.6 Modes colorimétriques

Afin de créer des images encore plus riches en couleurs (et donc disposer de plus qu'une palette limitée à 256 couleurs), l'idée de mélanger des couleurs primaires en « couches » est arrivée. Il faut savoir qu'il existe deux systèmes de représentation des couleurs par mélange, selon qu'on les reproduit sur un écran d'ordinateur ou sur support papier via une imprimante :

Mode couleur RVB : grâce au mélange des 3 couches de couleur, il est possible de reproduire un plus grand nombre de nuances qu'avec une palette en mode couleurs indexées.

Mode couleur CMJN : Comme les écrans d'ordinateur ne peuvent afficher que du RGB, Photoshop sépare les images CMJN en 4 couches (Cyan, Magenta, Jaune et Noir ou chaque couleur est exprimée en pourcentage) et converti le tout en RGB pour être affiché sur l'écran. Cependant pour l'utilisateur, le fichier possède bien 4 couches distinctes sur lesquels il est possible de travailler.

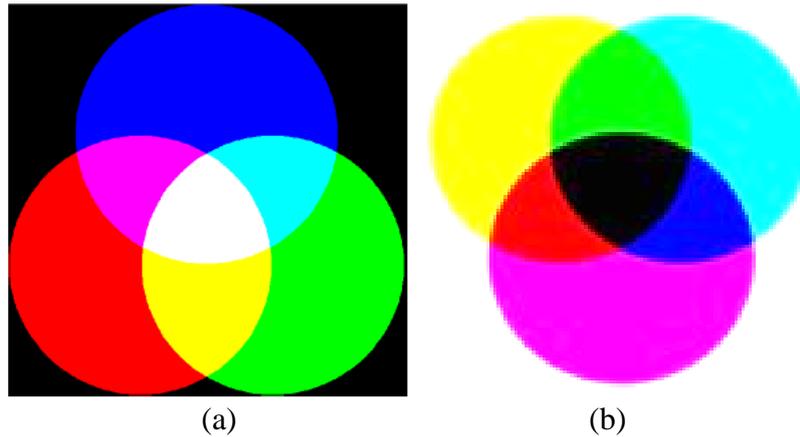


Figure 4 : Espace de couleur, (a) RVB, (b) CMNJ.

1.2.3 Prétraitement d'une image

Le traitement d'images est né de la nécessité de remplacer l'observateur humain par la machine. L'image ou les signaux provenant des capteurs ont alors été numérisés pour pouvoir être traités par l'ordinateur. Dans un deuxième temps, l'image a été codée et mémorisée sur différents supports (magnétique, magnéto-optique, etc.). La vision intervenant dans un grand nombre d'activités humaines, le champ des applications du traitement d'images est très vaste. À partir d'une image numérique, il convient d'extraire les informations pertinentes en regard de l'application concernée, les traiter puis les interpréter. Le terme générique d'analyse d'images désigne l'ensemble de ces opérations.

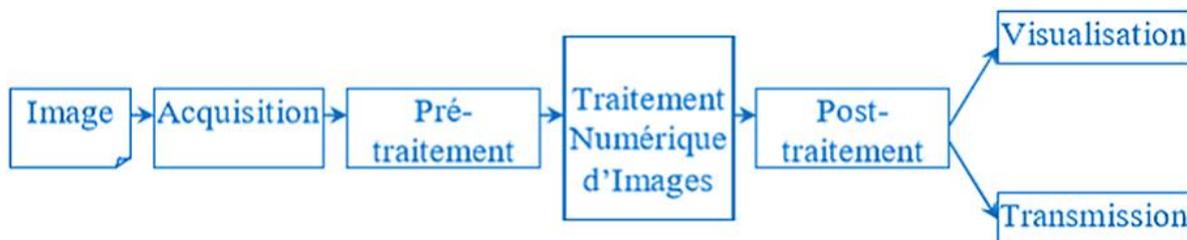


Figure 5 : Schéma d'un système de traitement d'image

Le traitement d'images est une discipline de l'informatique et des mathématiques appliquées qui étudie les images numériques et leurs transformations, dans le but d'améliorer leur qualité ou d'en extraire de l'information. Pour améliorer le contraste, nous avons appliqué légalisation d'histogramme (Nbp , nombre de pixels de l'image) :

- ✓ **Étape 1** : Calcul de l'histogramme $h(i)$ avec $i \in [0,255]$
- ✓ **Étape 2** : Normalisation de l'histogramme $h_n = h(i)/Nbp$ avec $i \in [0,255]$
- ✓ **Étape 3** : Densité de probabilité normalisé $c(i) = \sum h_n(j)$ avec $i \in [0,255]$
- ✓ **Étape 4** : Transformation des niveaux de gris de l'image.

1.2.3.1 Génération des caractéristiques

Cette étape représente le cœur du système de recherche, classification, détection, reconnaissance. On extrait de l'image les informations qui seront sauvegardées en mémoire pour être utilisées plus tard dans la phase de décision.

1.2.3.2 Normalisation des données

Généralement, les composantes d'un vecteur de caractéristique sont de nature hétérogène, qui pouvant influencer considérablement sur la qualité de la reconnaissance. Pour éviter ce problème, il faut normaliser les caractéristiques de sorte que leurs valeurs se situent dans des gammes similaires. Ou bien tout simplement par la normalisation des estimations de la N moyenne et de la variance.

1.2.3.3 Fusion d'information

L'opération de la fusion d'information est une solution adoptée pour pallier les limitations imposées individuellement par les descripteurs et leurs classifieurs correspondants. En général, une opération de fusion permet d'avoir une décision finale plus précise en choisissant convenablement la méthode de fusion [7]. Elle s'applique avant ou après l'opération de classification. Nous distinguons deux grandes catégories [8] : la fusion au niveau des descripteurs et la fusion au niveau de scores [9], [10]. Dans la première catégorie, il s'agit de concaténer les vecteurs de descripteurs pour les préparer à l'opération de classification. Pour cela, une étape de normalisation s'avère primordiale afin que les données des descripteurs soient définies dans le même intervalle. Dans la catégorie de fusion au niveau score, chaque classifieur propre à un descripteur opère indépendamment, ensuite les réponses sous forme de décisions ou de distances sont fusionnées.

1.3 Notion de texture

La texture peut être vue comme la représentation spatiale de motifs homogènes. Il se définit donc par une densité de répétitions de motifs, de formes et d'orientations. La texture se décrit en termes linguistiques tels que la rugosité, le contraste, la finesse, la régularité, etc., que les chercheurs se sont efforcés de caractériser depuis plusieurs années [11].

C'est ainsi que l'on trouvera dans les paragraphes suivants plusieurs définitions du contraste ou de l'homogénéité par exemple, qui toutes tendent à donner une définition mathématique du même phénomène visuel. Une texture présente, à une échelle donnée, le même aspect quelle que soit la zone observée. Dans ces conditions, on considère l'image comme la réalisation d'un processus stochastique stationnaire : c'est-à-dire que les statistiques calculées sont invariantes en translation. On rencontre deux types de définition de la texture :

La première est déterministe et fait référence à une répétition spatiale d'un motif de base dans différentes directions. Cette approche structurale correspond à une vision macroscopique des textures. C'est le cas, par exemple, des tissus ou de la peau de reptiles. Ainsi, B. Julesz, dès 1962, considère la texture comme l'agencement d'un motif de base qu'il appelle « texton » [12].

La deuxième est probabiliste qui cherche à caractériser l'aspect anarchique et homogène qui ne comprend ni de motif localisable, ni de fréquence de répétition principale (cas de l'herbe). Elle correspond à une vision microscopique. Gagalowicz [13] propose une synthèse des deux approches en considérant la texture comme « une structure spatiale constituée de l'organisation de primitives (ou motifs de base) ayant chacune un aspect aléatoire » [14].

1.3.1 Définition de texture

Donner une définition précise du terme "texture" est quelque chose de difficile. Il en existe un grand nombre, le référence [15], nous présentons un panel d'exemples :

– "Une région dans une image contient une texture constante si un ensemble de statistiques locales ou autres propriétés de la fonction image sont constantes, varient faiblement ou sont approximativement périodiques" [16].

– "Une image texturée est décrite par le nombre et les types de ses primitives tonales ainsi que leurs orientations spatiales. Elle ne peut pas être analysée sans une vue de la primitive tonale de référence. Pour certaines surfaces ayant un ton de gris flouté, il existe une échelle telle que la texture est inexistante. Au fur et à mesure que la résolution augmente, on observe une texture fine puis une texture grossière" [17]. Dans les travaux de [18], une définition perceptuelle de la texture est donnée :

– "Une texture est une région d'une image pour laquelle il est possible de définir une fenêtre de dimensions minimales, telle qu'une observation au travers de celle-ci se traduit par une perception (impression) visuelle identique pour toutes les translations possibles de cette fenêtre à l'intérieur de la région considérée ". Certaines de ces définitions décrivent les textures de façon perceptuelle, d'autres se focalisent plus sur une stabilité de mesures statistiques à l'intérieur d'une région ou sur la méthode de génération utilisée.

La plupart d'entre elles ne sont pas satisfaisantes dans le sens où elles proposent une description soit trop générale et donc imprécise, soit trop restrictive pour s'adapter à la diversité des cas rencontrés. Une texture est donc quelque chose d'abstrait, difficile à définir quantitativement de façon précise. Ce pendant une texture peut presque toujours être décrite

par des termes, des adjectifs souvent dépendants du domaine concerné (images médicales, images naturelles, matériaux, etc.) mais plus faciles à comprendre et donc plus faciles à manipuler.

Catégorie	Sous-catégorie	Méthodes
Statistiques	Propriétés del'histogramme	<ul style="list-style-type: none"> - Modèle binairede Gabor - GLCMet filtres de Gabor - Gaboret LBP - Transforméeenondeletteset GLCM - Variationd'énergie - Motifs binaires locaux etsélectionde points significatifs - Combinaison demodèleprimitifet caractéristiquestatistiques - Hybridcolor localbinarypatterns
	Matricedecooccurrence	
	Descripteurs binaires locaux	
	Basésur l'inscription	
	Energie de la texture	
Structural	Mesureprimitive	<ul style="list-style-type: none"> - Variation d'énergie - Détection de granularité de texture basée contours - Filtre morphologique - Squelette primitif et ondelettes.
	Caractéristiques decontours	
	Représentationdusquelette	
	Opérations morphologiques	
	SIFT	
Basémodèle	Autorégressive(AR) model	<ul style="list-style-type: none"> - Analyse multi-fractale dans la pyramide d'ondelettes multi-orientation - Modèles de texture de champ aléatoire de Markov - Modèles autorégressifs simultanés
	Modèles fractals	
	Modèle de champaléatoire	
	ModèleTexem	
Basée transformation	Spectral	<ul style="list-style-type: none"> - Modèle de Gabor binaire - Les canaux d'ondelettes et banque de filtres des canaux LL - GLCM et filtres de Gabor - Gabor et LBP - Transformée en ondelettes et GLCM - SVD et DWT - Squelette primitif et ondelettes
	Gabor	
	Wavelet	
	Transformation de Curvelet	

Tableau 1 : Catégorisation des méthodes de classification des textures. [19]

Domaines d'applications :

La texture de l'image nous donne beaucoup d'informations utiles sur le contenu de l'image, les objets à l'intérieur, arrière-plan, etc. L'analyse de texture touche la plupart des

domaines du traitement d'images, en particulier dans le processus d'apprentissage et l'extraction de caractéristique est en cours de discussion lorsque nous voulons comparer les images telles que [20-25] :

- ❖ Détection et reconnaissance de visage.
- ❖ Suivi des objets dans les vidéos.
- ❖ Diagnostics de la qualité des produits.
- ❖ Analyse d'images médicales.
- ❖ Télédétection.
- ❖ Végétation.

1.3.1.1 Familles de texture

Dans la littérature, il existe plusieurs familles de textures ; avec notamment les textures déterministes, les textures stochastiques et les textures observables [26]. Les textures déterministes se caractérisent par la répétition d'un motif géométrique aussi appelé "texton". Ce genre de textures peut donc facilement être synthétisé à partir d'un motif si on connaît son orientation et sa taille.

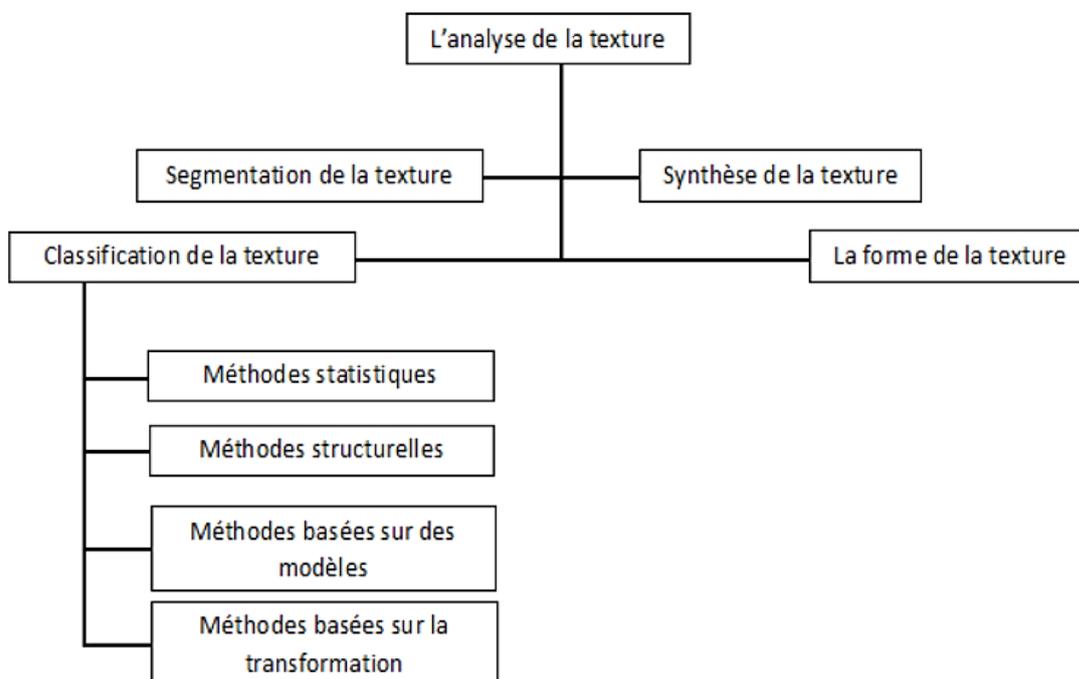


Figure 6 : Vue générale de l'analyse de texture.

Au contraire, les textures stochastiques sont irrégulières. Elles obéissent à certaines lois statistiques et peuvent être considérées comme une réalisation d'un champ aléatoire bidimensionnel. Une texture observable se décrit comme un mélange entre une texture stochastique et une texture déterministe.

En effet, à l'intérieur de ce type de textures, nous pouvons trouver une répétition de motifs très similaires mais pas identiques.

Certains auteurs n'utilisent pas le vocabulaire précédent et préfèrent parler de textures structurées (macro-textures) ou de textures aléatoires (micro-textures) [27]. Les textures structurées ont pour caractéristique la répétition de primitives à intervalles réguliers. Parmi elles, nous pouvons différencier les textures parfaitement périodiques (textures déterministes). Les textures dont les primitives ont subi quelques transformations (textures observables). Les textures aléatoires (textures stochastique) se distinguent par un aspect plus fin (sable, herbe, etc.). Il est difficile d'en isoler une primitive de base et d'identifier une fréquence de répétition. Il existe une grande diversité de textures [28].

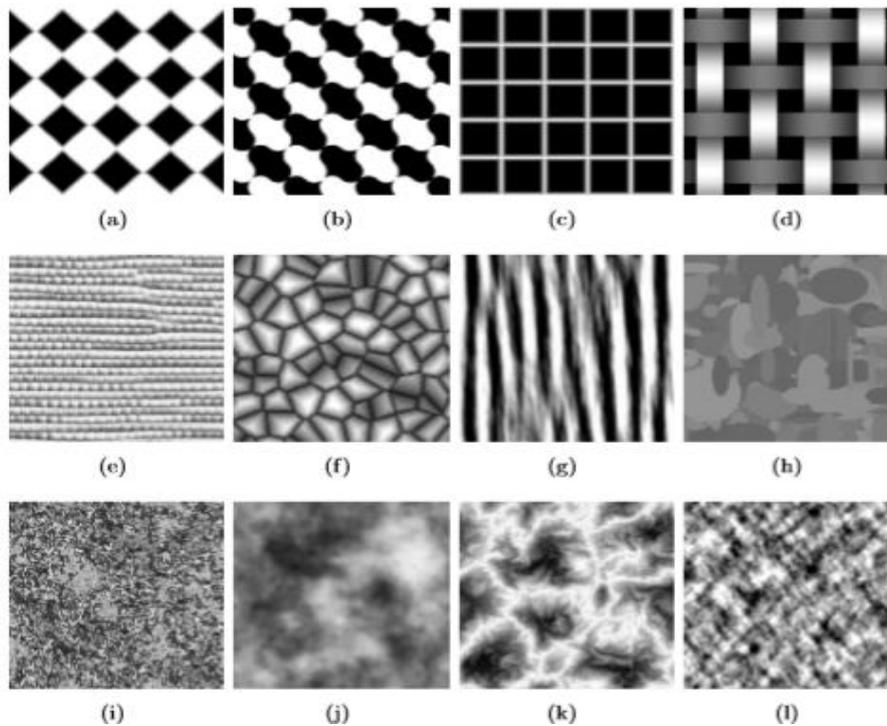


Figure 7 : Familles de textures : [a-d] Déterministes, [e-h] Observables, [i-l] Stochastiques.

1.4 Conclusion

Dans ce chapitre nous sommes particulièrement intéressés aux différentes représentations des images ainsi que leurs caractéristiques. Nous avons aussi présenté le principe de l'analyse de la texture en vision par ordinateur et son application dans le domaine de la classification de texture. Ensuite dans le chapitre suivant on va présenter les descripteurs orientés de traitement et d'extraction des caractéristiques de la texture. On va concentrer sur les descripteurs les plus populaires et les plus utilisés dans le domaine de classification des texture.

Chapitre 2

‘Apprentissage Automatique’

2.1 Introduction

Les méthodes de classification supervisée abordées précédemment sont divisibles en deux phases. La première est une phase d'extraction de caractéristiques durant laquelle on extrait pour chaque objet de la base d'apprentissage les propriétés discriminantes. Puis, la seconde est une phase d'apprentissage où un classifieur apprend les meilleures frontières entre les différentes classes. Cependant le type de caractéristiques extraites n'est pas forcément optimal pour le problème traité et est choisi indépendamment du classifieur. Une solution pour éviter ce problème est d'effectuer un apprentissage profond (Deep Learning) où les phases d'extraction de caractéristiques et de classification sont jointes et se réalisent simultanément. Il y a d'autres méthodes se basant sur la classification des caractéristiques par un seul classificateur comme SVM, KNN, K-means, toutes ces méthodes consistent à déterminer le modèle le plus proche d'une image d'entre parmi les images de la base de données.

Methods de reconnaissance		BD de test	Method	Avantages	Inconvénients	Résultat
Apprentissage auto						
➤ ACP	Sirovich et al.	Yale, ORL, CMU	PCA	Principe simple	Précision faible	78% - 88.5% - 75.7%
➤ SVM	Osuna et al.	BD Propre	SVM	Large classes d'images	Instabilité des calculs	74.2%
➤ SNoW	Yang et al.	ORL, CMU	Windows	Meilleures performances		87.24% - 87.52%
➤ Bayésien	Moghaddam et al.	FRET	Probabilité	Réduire la dimension	Instables chang d'illumination	85.42%
➤ HMM	Bevilacqua et al.	ORL	HMM, CNN	Robustesse aux expressions	Temps de calculs importants	99.50%
➤ WT-CNN	Vankayalapati et al.	FRET, ORL	Wavelet Trans	Amélioration de précision	N'est pas invariants la pose	89.27% - 90.02%

Tableau 2: Méthodes d'apprentissage automatique.

2.2 Systèmes de reconnaissance de forme

Les systèmes classiques de reconnaissance de forme sont divisés en deux modules principaux:

- Un module d'extraction de caractéristiques (LBP, HOG, ...etc.).
- Un module de classification (SVM, K-NN, ...etc.). [32]

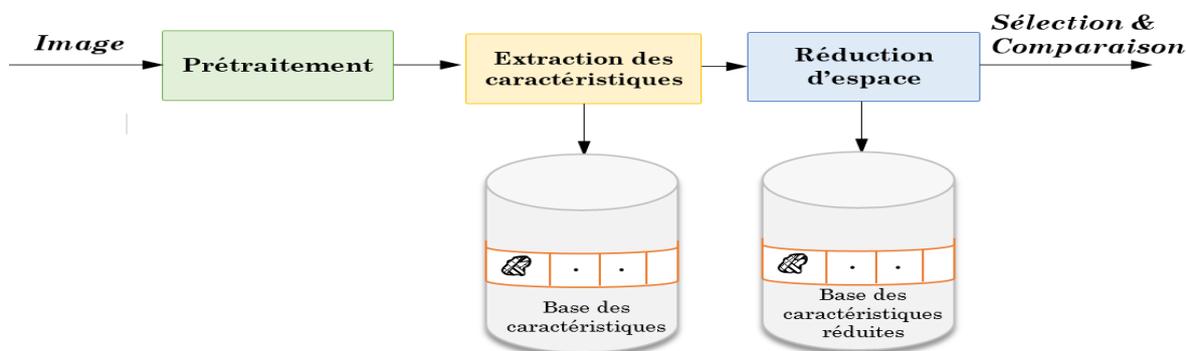


Figure 8 : Étapes d'extraction des caractéristiques.

Chaque approche contient différents modèles pour obtenir la reconnaissance des expressions faciales et l'analyse de texture, nous allons mettre en évidence certaines de ces méthodes.

2.2.1 Extraction des caractéristiques

Une étape préliminaire et très importante est l'apprentissage du classificateur. Il s'agit d'entraîner le classificateur afin de le sensibiliser à ce que l'on veut détecter, ici des visages. Pour cela, il est mis dans deux situations : La première où une énorme quantité de cas positifs lui sont présentés et la deuxième où, à l'inverse, une énorme quantité de cas négatifs lui sont présentés. Concrètement, une banque d'images contenant des visages de personnes est passée en revue afin d'entraîner le classificateur. Ensuite, une banque d'images ne contenant pas de visages humains est passée. Dans le cas présent, Viola et Jones ont entraîné leur classificateur à l'aide d'une banque d'images du MIT. Il en résulte un classificateur sensible au visage humain. Il se présente sous la forme d'un fichier XML. Dans l'absolu, on serait en mesure de détecter n'importe quel signe distinctif à partir d'un classificateur entraîné à cela.

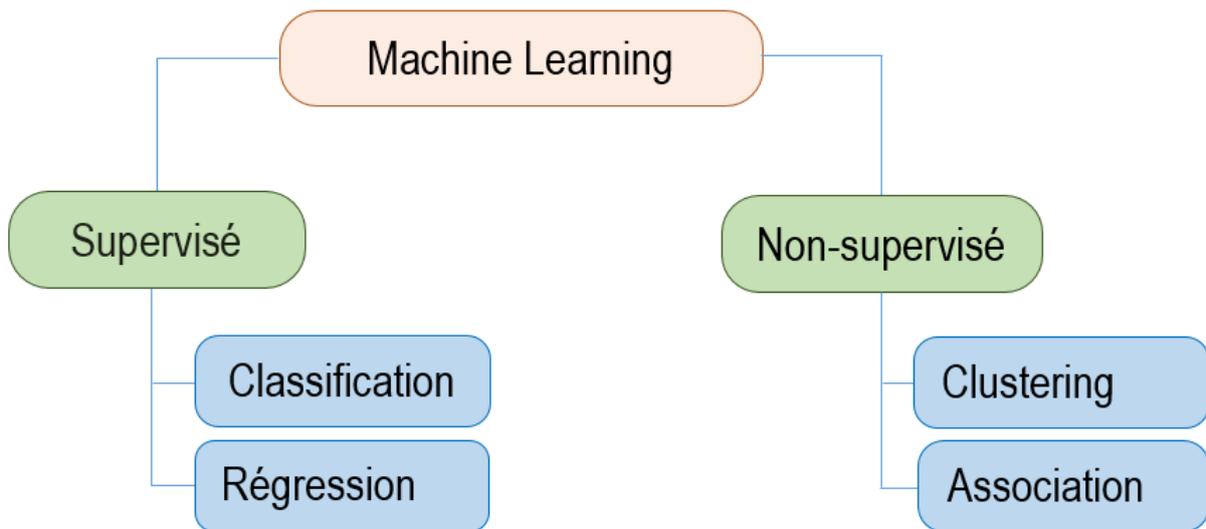


Figure 9 : Classification des méthodes d'apprentissage automatique.

Dans le domaine de l'apprentissage automatique, il existe deux principaux types de tâches : supervisées et non supervisées [42]. L'apprentissage supervisé est la technique permettant d'accomplir une tâche en fournissant aux systèmes des modèles d'apprentissage, d'entrée et de sortie. Tandis que l'apprentissage non supervisé est une technique d'auto-apprentissage dans laquelle le système doit découvrir par lui-même les caractéristiques de la population d'entrée [42]. Pour mieux comprendre voici quelques exemples :

Méthode non supervisée : Un client vient d'acheter un livre sur Amazon. Je souhaite retrouver les personnes avec les mêmes goûts pour orienter ce client. On ne connaît pas à

l'avance les classes possibles. La base d'apprentissage est composée de clients ayant fait des achats. On souhaite regrouper les clients ayant les mêmes goûts dans une classe.

Méthode supervisée : On reçoit une image binaire et associer à une des 26 lettres de l'alphabet. On connaît aux classes possibles. La base d'apprentissage est étiquetée avec ces 26 classes.

2.2.2 Descripteurs de texture

Un descripteur a pour rôle de caractériser la discriminabilité ou distinctivité d'une zone identifiée par un point : cela se construit à partir d'informations discriminantes de la zone autour du point d'intérêt. Dans une même image, les points d'intérêt doivent pouvoir se distinguer les uns des autres par leur signature spécifique. Là encore, plusieurs descripteurs de points d'intérêt sont abordés. Nous ferons un focus sur les descripteurs pouvant s'utiliser en temps réel ou proche temps réel. Certains de scripteurs sont indissociables du détecteur utilisé :

- ❖ Descripteurs non-binaires
- ❖ Descripteurs binaires

Il existe trois grandes familles de descripteurs :

- **Descripteurs photométriques :**
 - ❖ Descripteurs par histogrammes
 - ❖ Descripteurs de moments
- **Descripteurs de textures :**
 - ❖ Descripteurs par co-occurrence
 - ❖ Descripteurs par motifs binaires locaux
- **Descripteurs de formes par analyse du gradient :**
 - ❖ Histogramme de Gradient Oriente

L'histogramme de l'image en niveau de gris est considéré comme un descripteur de référence.

a)- Histogramme de l'image de dentelle :

L'histogramme est défini comme une fonction discrète qui associe à chaque valeur niveau de gris le nombre de pixels prenant cette valeur. Les images de dentelles sont des images couleurs (ou gris). Elles ont été transformées en images en niveaux de gris.

b)- Histogramme des motifs binaires locaux :

L'opérateur des motifs binaires locaux (LBP) a été proposé à la fin des années 90 par Ojala [5], [6]. L'idée de cet opérateur de texture est d'assigner à chaque pixel un code dépendant des niveaux de gris de son voisinage. Le niveau de gris du pixel central (i_c) de coordonnées (x_c, y_c) est comparé à celui de ses voisins (i_n).

Autres descripteurs se basant sur la spécificité des images de dentelles qui présentent des textures similaires. Ces descripteurs, qui ont prouvé leurs capacités discriminantes pour l'information de texture [5], [6], [29], [30], sont :

- ❖ **SIFT** : Transformation de Caractéristiques Visuelles Invariantes à l'Échelle
- ❖ **HOG** : Histogramme de Gradient Orienté
- ❖ **LBP** : Méthode du Motif Binaire Local
- ❖ **MLBP** : Méthode du Motif Binaire Local Modifié
- ❖ **LTP** : Modèle Ternaire Local
- ❖ **LDP** : Modèle Directionnel Local
- ❖ **LDN** : Modèle Directionnel Number
- ❖ **LGC** : Local Codage Gradient
- ❖ **LGP** : Local Gradient Pattern
- ❖ **LPQ** : Quantification par Phase Locale
- ❖ **WLD** : Weber Local Descriptor ... etc.

Il existe plusieurs techniques d'extraction des caractéristiques, nous avons adopté sur deux méthodes : LBP, HOG.

2.2.3 Méthodes d'extraction des caractéristiques

La génération des caractéristiques est une étape cruciale dans tout système de reconnaissance. Dans [31], les auteurs présentent un état de l'art sur les méthodes d'analyse de textures 2D à l'intérieur duquel ils proposent une classification de ces techniques selon quatre familles :

- ❖ Les méthodes statistiques
- ❖ Les méthodes géométriques
- ❖ Les méthodes basées sur un modèle
- ❖ Les méthodes empruntées au traitement du signal

Généralement Les méthodes de reconnaissance et extraction des caractéristiques peuvent être classées en trois grandes approches :

- ❖ **Méthode globale** : dans laquelle on analyse l'image (l'image pixellisée) dans son entier.
- ❖ **Méthode locale** : basé sur un modèle, dans laquelle le système essaie de détecter, regrouper et reconnaître les différents éléments constitutifs de l'image dans le visage.
- ❖ **Méthode hybride** : c'est une méthode qui combine les deux approches précédentes.

Ci-dessus on présente une comparaison entre les méthodes d'extraction des caractéristiques :

	Vitesse de réponse	Complexité d'algorithme	Réponse
Globale	Très bien	Simple	Médiocre
Locale	Moyen	Complexe	Bien
Hybride	Mauvais	Très complexe	Très bien

Tableau 3 : Tableau de comparaison entre les méthodes d'extraction des caractéristiques.

Algorithme	Type de Caractéristique	Rotation	Bruit	Luminosité	Échelle
FAST	Coin	Mauvais	Moyen	Mauvais	Moyen
Harris	Coin	Mauvais	Moyen	Moyen	Mauvais
CANNY	Bords	Moyen	Très Bien	Bien	Mauvais
ROBERTS	Bords	Moyen	Mauvais	Bien	Mauvais
SOBEL	Bords	Moyen	Mauvais	Bien	Mauvais
GABOR	Bords / Texture	Très Bien	Bien	Très Bien	Très Bien
LBP	Texture	Tres Bien	Mauvais	Très Bien	Mauvais

Tableau 4: Tableau de comparaison entre les algorithmes d'extraction des caractéristiques.

2.2.3.1 Motif binaire local 'LBP'

La méthode LBP (Local Binary Pattern) est l'un des techniques qui a récemment attiré l'attention dans la représentation des expressions faciales et d'analyse de la texture, a été proposée par Ojala et al, en 1996. LBP basé sur la séparation de l'image en plusieurs petites régions pour extraire la caractéristique de chaque point ou bien pixel. L'opérateur LBP travaille parmi huit voisins de pixel, en utilisant la valeur de pixel centrale comme seuil. Lorsque le pixel voisin a une valeur de gris supérieure au pixel central par rapport à la valeur attribuée à ce pixel, on obtient "1". Le même cas se produit lorsque les valeurs sont identiques. Ensuite, le code LBP est créé par concaténation de zéros et des uns en un code binaire ou en huit valeurs pour le pixel central [17]. LBP est un moyen puissant de description de texture et parmi ses propriétés dans des applications réelles sont ses discriminatives puissances, simplicité de calcul.

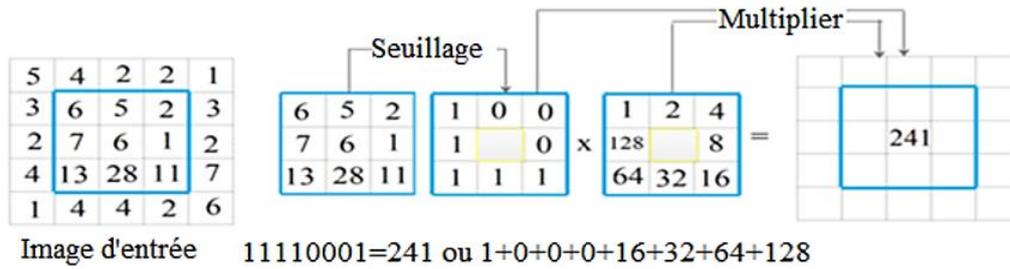


Figure 10 : Illustration de LBP basique.

LBP a été étendu ultérieurement basant sur des voisinages de taille différente (voisinage de P pixels différentes c-à-dire différentes échelles) et la forme circulaire de rayon R. La majorité des recherches qui était basée sur la méthode de LBP pour améliorer la reconnaissance des expressions faciales, donnent une précision entre 91.7% et 95.8%.

Une expérience basée sur la combinaison de (LBP + SVM) donne une efficacité de 90% [17] [33], une autre expérience utilisant un modèle LBP de pièces du visage et réseau de neurones multicouches donne une efficacité de 95.8% [34]. L'un des inconvénients de l'opérateur fondamental de LBP est qu'il est incapable de saisir certaines caractéristiques dominantes. Afin de pouvoir traiter la texture à des échelles complètement différentes, il a donc fallu travailler sur l'opérateur et ainsi émerger l'utilisation de voisinage de différentes tailles. Un voisinage local est défini comme un groupe de points d'échantillonnage de la même dimension répartis sur un cercle et centrés à l'intérieur des pixels. Les interpolations sont interpolées à l'aide d'une interpolation bilinéaire, permettant ainsi de définir tout rayon et toute variété de points d'échantillonnage dans le voisinage [35].

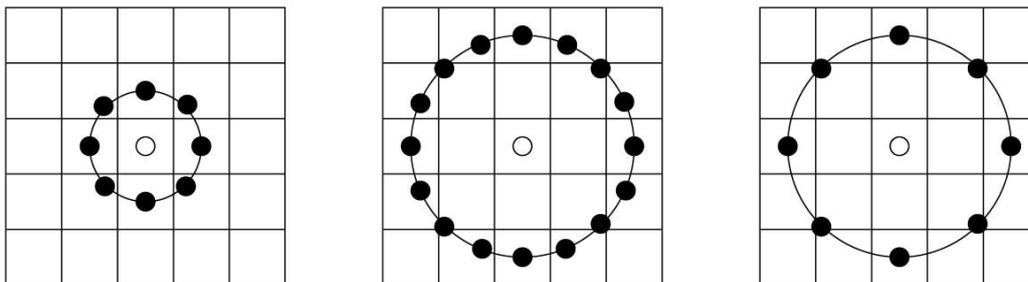


Figure 11 : Exemples de voisinages avec différentes valeurs de (P, R).

Pour les échantillons de l'opérateur LBP étendu, où la notation (P, R) désigne un voisinage de P à l'intérieur duquel les points sont équidistants (la même dimension) sur un cercle de rayon R créant un ensemble voisin de symétrie circulaire [35].

Nous disons que le modèle binaire local est uniforme lorsque nous avons aux plus deux transitions binaires de 0 à 1 ou vice versa par exemple 00000000 (0 transitions), et

01110000 (2 transitions) sont uniformes, le modèle binaire 11001001 (4 transitions), et 01010011 (5 transitions) n'est pas uniforme.

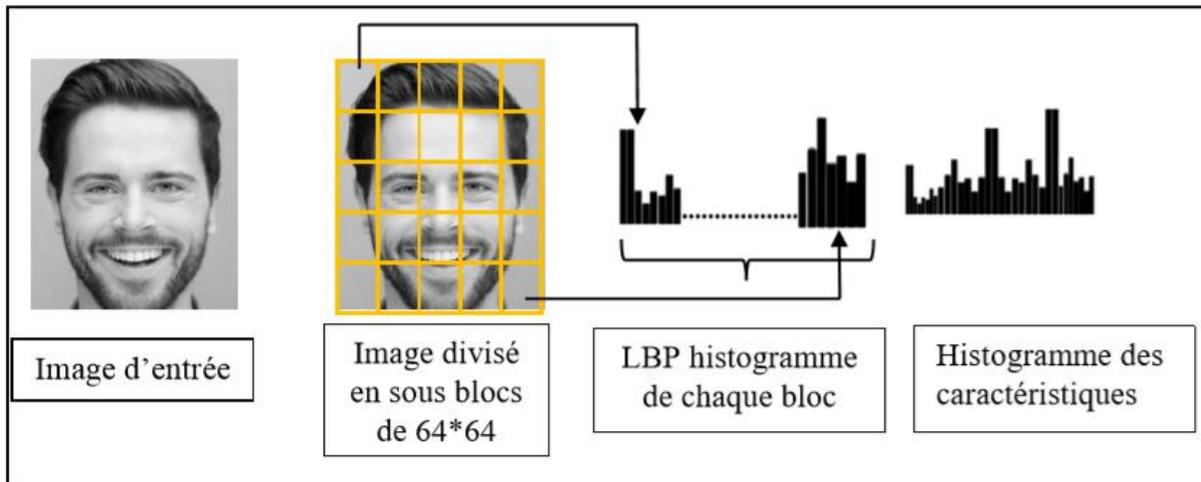


Figure 12 : Schéma d'obtention d'un histogramme LBP.

L'opérateur LBP produit $2p$ ($p = 8$) valeurs de sortie, nous avons donc un vecteur de 256 valeurs. Dans [36] Ojala et al, ont observé que bien que seulement 58 des 256 motifs à 8 bits soient uniformes, près de 90% de tous les voisinages d'image observés sont uniformes et bon nombre des motifs restants contiennent essentiellement du bruit. Ainsi, l'histogramme de chaque motif uniforme est réduit de 256 à 59 valeurs. Récemment le LBP a été développé avec de nombreuses variantes pour améliorer les performances dans différentes applications. Ces variations se concentrent sur différents aspects de l'opérateur LBP d'origine [36] :

- ❖ Amélioration de sa capacité de discrimination.
- ❖ Amélioration de sa robustesse.
- ❖ Sélection de son voisinage.
- ❖ Extension aux données 3D.
- ❖ Combinaison avec d'autres approches.

Il y a d'autres extensions de LBP tels que : ILBP, CLBP, VLBP, ELBP, MLBP, CS-LBP, OC-LBP, LBP-TOP, Soft LBP...etc.

2.2.3.2 Histogramme de gradient orienté 'HOG'

L'un des méthodes de reconnaissance des expressions faciales c'est HOG (Histograms of oriented gradients) qui a été proposée par Dalal et Triggs 2005, [37]. Les descripteurs HOG sont caractérisés par deux principaux paramètres, la taille de la cellule et le nombre de cases d'orientation. L'implémentation de ces descripteurs peut être obtenue en divisant l'image en petites régions connectées (cellules) pour lesquelles un histogramme des directions de

gradient est calculé. La combinaison de ces histogrammes représente alors le vecteur de caractéristiques.

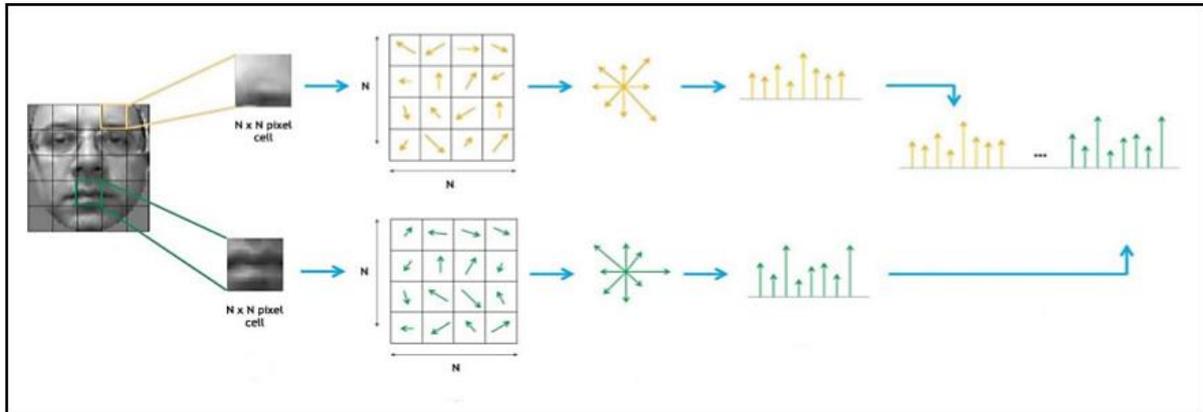


Figure 13 : Exemples d'histogramme obtenue dans méthode de HOG.

L'algorithme implique différentes étapes, la première étape est le calcul des valeurs du gradient. La deuxième étape consiste à créer les histogrammes de cellules. Chaque pixel de la cellule génère un vote pondéré pour un canal d'histogramme basé sur l'orientation en fonction des valeurs trouvées dans le calcul du gradient.

Les cellules sont regroupées en blocs plus grands, liés dans l'espace. Le descripteur HOG est alors le vecteur qui concatène les composants des histogrammes normalisés de tous les blocs les régions. Ici, plusieurs paramètres doivent être pris en compte : la taille de la cellule en pixels, le nombre de cellules par bloc et le nombre d'orientations [38]. Le taux de précision de la méthode HOG et son efficacité est environ 95.8%, à partir d'une expérience basée sur la méthode de HOG, [37].

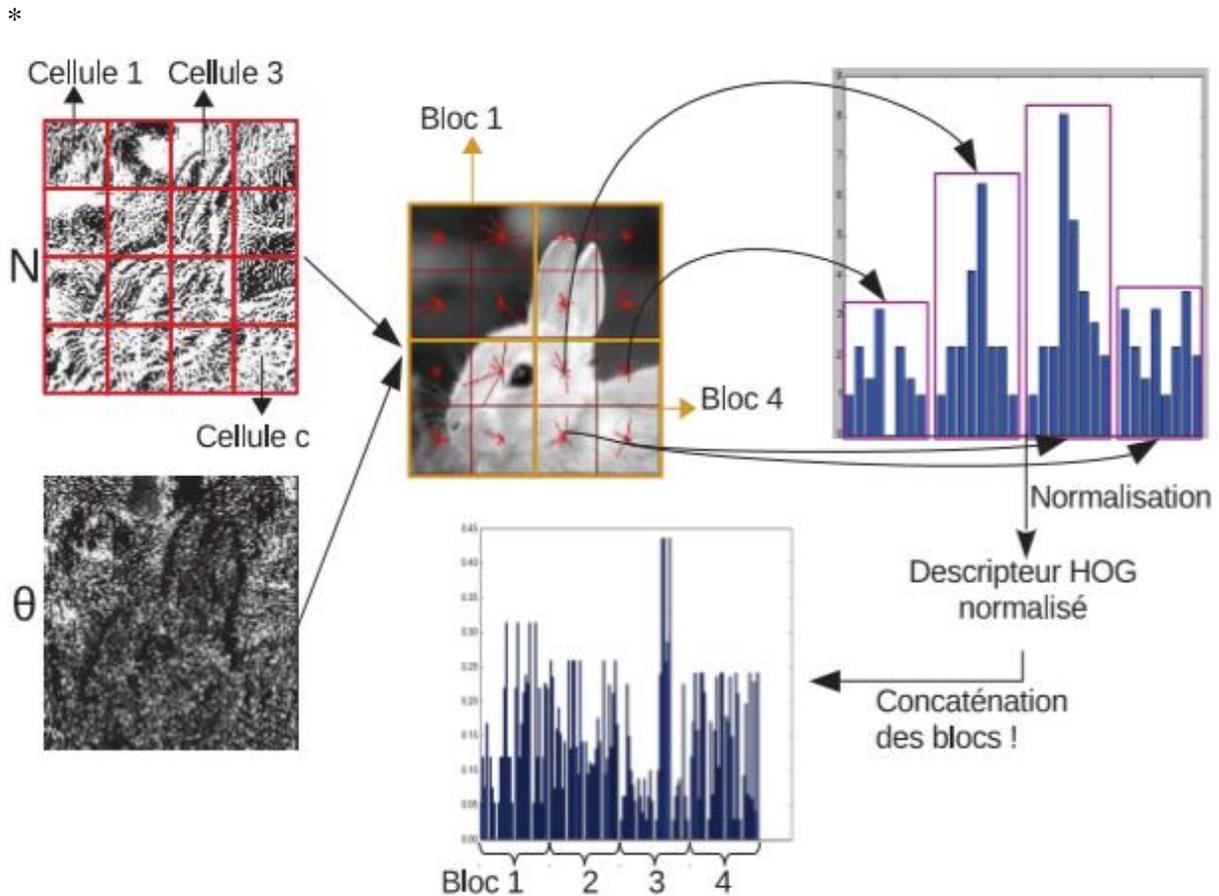


Figure 14 : Expérience basée sur la méthode de HOG.

Pour de meilleurs résultats, les histogrammes locaux sont normalisés en contraste, en calculant une mesure de l'intensité sur des zones plus larges que les cellules, appelées des blocs, et en utilisant cette valeur pour normaliser toutes les cellules du bloc. Cette normalisation permet une meilleure résistance aux changements d'illuminations et aux ombres. C'était brièvement le concept général de descripteur de HOG, pour l'obtention d'histogramme finale il faut passer par cinq étapes nécessaires comme suite [39] :

a) Calcul du gradient

Le calcul rapide du gradient peut être fait, par exemple, par des masques de dérivation simple 1D centré. On applique un masque $[-1 \ 0 \ 1]$ puis un masque $[-1 \ 0 \ 1]^T$ ou non centré $[-1, 1]$. Il y a d'autre type de masquage tels que filtre de Sobel (3*3), ou Derich ou, filtrage gaussien pour dimension 2D.

b) Construction de l'histogramme

Dans cette étape l'image est divisée en 8*8 cellules et on calcule l'histogramme d'orientation de gradient pour chaque cellule, qui vote alors pour une classe de l'histogramme, en fonction de l'orientation du gradient à ce point. Le vote du pixel est pondéré par l'intensité

du gradient en ce point. Les histogrammes sont uniformes de 0 à 180° (cas non signé) ou de 0 à 360° (cas signé) [39].

c) Formation et normalisation des blocs

Dans l'étape précédente, nous avons créé un histogramme basé sur le calcul de l'orientation de gradient de l'image. Les valeurs des HOG peuvent varier de manière significative à cause de la sensibilité à l'éclairage (variation). Cela change les valeurs de l'histogramme, et pour cela nous avons fait une normalisation sur l'histogramme.

La normalisation se fait en formant des blocs, chacun est constitué de quatre cellules de 6x6 pixels, ou effectue la normalisation sur ces blocs. Il existe une corrélation entre le nombre de cellules de bloc et la précision de descripteur, si le nombre de cellules dans un bloc est élevé, le descripteur est plus discriminant. Mais la taille des cellules est alors plus petite. La taille des cellules doit cependant rester assez importante car, si les cellules sont trop petites, les informations qu'elles contiennent sont trop peu nombreuses et donc peu pertinentes, de plus, un grand nombre de blocs donne un résultat final plus précis [40]. Il y a d'autres extensions de HOG tels que : PHOG, UHOG, EXHOG, HOG + SVM ...etc.

2.3 Classification automatique

De manière générale, les systèmes d'apprentissage automatique sont des systèmes de classification ou de reconnaissance de formes. Un système d'apprentissage automatique est une boîte noire avec une entrée, par exemple une image, un son, ou un texte, et une sortie qui peut représenter la classe de l'objet dans l'image, la parole ou le sujet dans le texte [42].

Les internautes contribuent depuis de nombreuses années à former des systèmes d'apprentissage automatique. Les premiers programmes d'apprentissage automatique servent à identifier un chat, un panneau STOP ou un feu de circulation. Bien sûr, au début l'AI fait à peine la différence entre l'image d'un chat et celle d'un chien, pour que le programme comprenne ce qui est un chat, un superviseur apprend au programme que cette image contient un chat. Mais pour progresser, il faut développer le bon algorithme d'apprentissage, et cela prend du temps très important. Aujourd'hui l'apprentissage automatique se développe de plus en plus particulièrement dans les systèmes experts ou les super calculateurs avec le grand saut qu'il a fait grâce aux réseaux de neurones artificiels et au Deep Learning [43].

2.3.1 Classification supervisée

L'apprentissage supervisé, dans le contexte d'IA, est un système qui fournit les entrées et les sorties attendues. Les données en entrée et en sortie sont étiquetées pour la classification, afin d'établir une base d'apprentissage pour un traitement ultérieur des données [42].

L'évaluation supervisée consiste à comparer le résultat fourni par un algorithme avec une segmentation de référence (vérité-terrain). Cette dernière peut être une image de synthèse ou une image construite à la main par un expert du domaine de l'application dans le cas d'images naturelles, en traçant des contours (par des médecins, des géographes, etc.), en utilisant des outils informatiques de dessin. Malheureusement, l'évaluation sur images naturelles n'est pas toujours facile à exécuter, parce qu'il n'existe pas de solution unique à la division d'une image en régions adéquates, autrement-dit, la division de l'image en régions est une notion qui dépend absolument de l'application. On peut cependant considérer que la différence entre deux divisions humaines d'une même image est une question de différence de niveau de détails, c'est-à-dire qu'une division peut être une sur-segmentation de l'autre et inversement [49].

Deux manières sont envisageables pour l'évaluation d'une segmentation machine par rapport à une segmentation de référence, la première consiste à utiliser les régions dont on trouve les critères GCE (Global Consistency Error), PRI (Probabilistic Rand Index), et VoI (Variation of Information). Ces critères s'intéressent plus particulièrement à la comparaison des contenus des régions, dont le but est de rechercher la correspondance entre les pixels dans les segmentations à comparer. L'autre manière consiste à calculer la distance entre les segments qui forment les frontières des régions autre critère BDE (Boundary Displacement Error) [49]

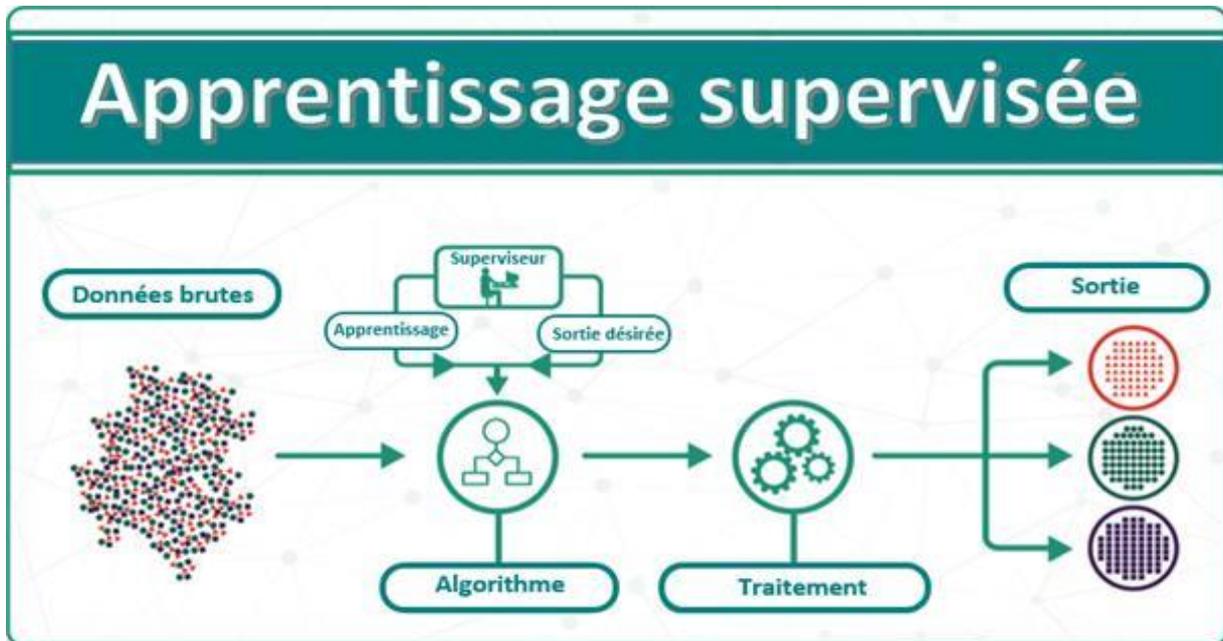


Figure 15 : Schéma général d'un système d'apprentissage supervisé [44].

Dans cette méthode de classification, on dispose déjà d'exemples dont la classe est connue et étiquetée. Une information sur les données à traiter est disponible et est utilisée

pour entraîner le processus de classification, cela constitue la phase d'apprentissage du modèle. Cette information appelée ensemble d'apprentissage est généralement constituée d'un ensemble d'individus {caractéristiques, classe associée}. Dans le cas de la classification de textures, l'ensemble d'apprentissage est constitué d'un ensemble de textures "types".

Chaque individu est donc composé du couple (caractéristiques de textures, texture associée). Cet ensemble est alors appris par un algorithme de classification supervisée classique parmi lesquels, on cite : les k-plus proches voisins (k-NN), les réseaux de neurones, les machines à support de vecteurs (SVM), etc. Une fois la phase d'apprentissage réalisée, l'algorithme de classification est alors utilisé afin de déterminer la classification d'un ensemble d'individus tests composé d'un grand nombre d'échantillons. Cette approche de classification de textures supervisée reste un domaine de recherche très actif. Les problèmes d'apprentissage supervisés peuvent être classés en problèmes de classification et de régression [48] :

- ❖ **Classification** : le problème de classification est un problème où nous utilisons des données pour prédire dans quelle catégorie l'objet en question appartient. Un exemple de problème de classification pourrait être l'analyse d'une image pour déterminer si elle contient une voiture ou une personne, ou l'analyse de données médicales pour déterminer si une certaine personne fait partie d'un groupe à haut risque pour une certaine maladie ou non.
- ❖ **Régression** : les problèmes de régression, par contre, sont des problèmes où nous essayons de faire une prédiction sur une échelle continue. Des exemples pourraient être de prédire le cours des actions d'une entreprise ou de prédire la température de demain sur la base de données historique.

La principale différence entre la régression et la classification est que la variable de sortie dans la régression est numérique (ou continue) et pour la classification est catégorielle (ou discrète).

2.3.1.1 Séparateurs à vaste marge 'SVM'

Le séparateur à vaste marge (SVM) est un algorithme permettant de trouver la meilleure séparation entre deux ensembles de données. Pour cela, nous cherchons dans une dimension donnée le meilleur hyperplan séparateur séparant deux jeux de données. Afin de trouver une séparation binaire le SVM calcule un vecteur appelé poids qui est orthogonal au plan séparateur.

Le choix de la fonction de perte est important et dépend du problème. Elle permet de régler le degré d'ajustement des données en mesurant la différence entre la sortie prédite et la valeur attendue. Une des fonctions de régression la plus connue [18] est la fonction logistique.

Les SVM linéaires cherchent à résoudre le problème sans effectuer de transformations à l'aide de l'espace de re description. Ils possèdent donc une complexité de l'ordre de $O(N)$. Ils sont utilisés pour traiter les grands volumes de données avec plusieurs millions de descripteurs et sont particulièrement efficaces pour les problèmes dit spartes [41].

L'objectif de la classification supervisée est d'établir un ensemble de règles formant un modèle, qui vont permettre de définir des frontières entre des objets de natures différentes. Pour cela, la première étape consiste à décrire les objets de manière fine. La seconde étape consiste à construire un modèle, partir d'objets connus, c'est à dire d'objets dont nous connaissons à l'avance la nature. La nature d'un objet est appelée la classe, le label ou encore l'étiquette.

Cette méthode de classification de données a été proposée par Vapnik en 1995 [46]. Initialement, elle a été définie pour un problème de discrimination à deux classes (cas binaire). De manière simple, il s'agit de construire, dans l'espace des attributs un hyperplan séparant les points représentatifs des prototypes de chacune des deux classes. Cette approche, également appelée méthode des séparateurs à vastes marges, repose sur deux notions clés : la notion de marge maximale et la notion de noyau.

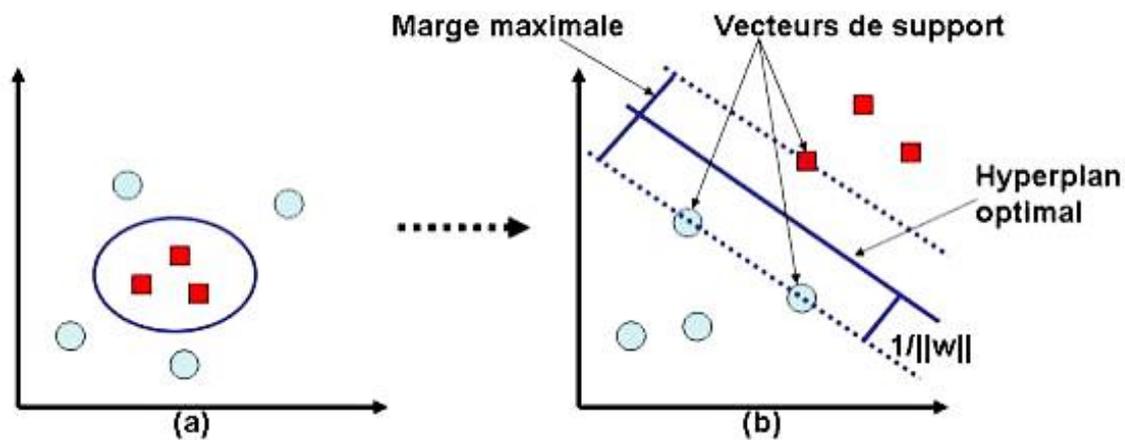


Figure 16 : Illustration du principe des SVM.

Il existe deux classes différentes classées à l'aide d'une frontière de décision ou d'un hyperplan. SVM (Support Vector Machine) peut être classifié en deux types : **linéaire** et **non linéaire**. La marge est la distance séparant la frontière de séparation (l'hyperplan) et les prototypes les plus proches de cette frontière. Ces prototypes sont appelés vecteurs supports. La frontière de séparation est choisie comme étant celle qui maximise la marge [77].

- **SVM linéaire** : est utilisé pour les données linéairement séparables, ce qui signifie que si un ensemble de données peut être classé en deux classes en utilisant une seule ligne droite, alors ces données sont appelées données linéairement séparables, et le classificateur est utilisé comme classificateur SVM linéaire [52].
- **SVM non linéaire** : est utilisé pour les données non linéairement séparables, ce qui signifie que si un ensemble de données ne peut pas être classé en utilisant une ligne droite, ces données sont qualifiées de données non linéaires et le classificateur utilisé est appelé classificateur SVM non linéaire [52].

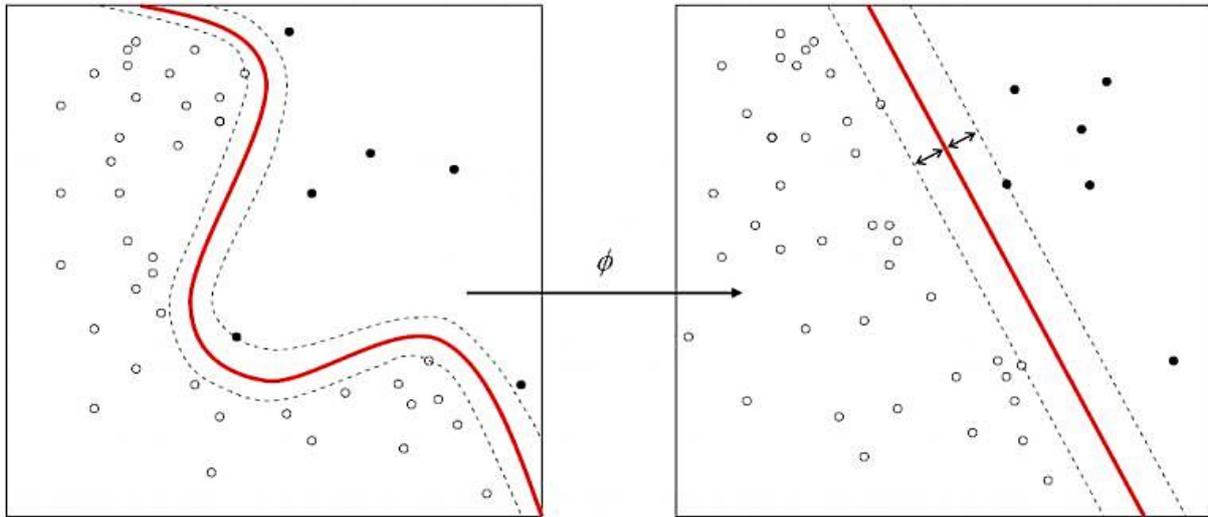


Figure 17: SVM linéaire et non linéaire [53].

Parmi les modèles des SVM, il y a les cas linéairement séparables et les cas non linéairement séparables. Les premiers sont les plus simples car ils permettent de trouver facilement le classifieur linéaire. Pour surmonter les inconvénients des cas non linéairement séparables, l'idée des SVM est de changer l'espace des données. La transformation non linéaire des données peut permettre une séparation linéaire des exemples dans un nouvel espace.

Il s'agit d'une transformation d'un problème de séparation non linéaire dans l'espace de représentation en un problème de séparation linéaire dans un espace de re-description de plus grande dimension. Cette transformation non linéaire est réalisée via une fonction noyau. En pratique, quelques familles de fonctions paramétrables sont connues et il revient à l'utilisateur de SVM d'effectuer des tests pour déterminer celle qui convient le mieux pour son application. On peut citer les exemples suivants : Polynomiale, Gaussien, Sigmoidale et Laplacien.

Pour un problème de discriminations multiples, dans de nombreux cas comme la classification des visages et des expressions faciales, « un contre tous » et « un contre un »

sont les deux approches les plus utilisées pour la construction de SVM multi-classes à partir d'un ensemble de SVM binaires. La première se base sur le principe du vote majoritaire, dans lequel la classe est assignée par le SVM ayant le plus grand score en sortie.

Approche « un contre un » : consiste à utiliser un SVM par couple de catégories. Chaque fois le SVM assigne une instance à l'une des deux classes, la classe de vote majoritaire sera bénéficié d'un en plus. La classe avec un maximum de votes détermine l'étiquette de l'instance [54].

Approche « un contre tous » : peut souffrir de la répartition déséquilibrée des exemples positifs par rapport aux exemples négatifs. C'est la raison pour laquelle, dans notre travail dans le cadre de cette thèse, nous nous servons de la stratégie « un contre un » pour résoudre le problème de classification multiple de visages expression faciale [54].

Avantages :

- Grande degré de précision de prédiction
- Classification d'images fiable et efficace.
- Manipule de grandes quantités de données.
- Facile et pratique avec un petit jeu de données.
- Elle est justifiée et fondée théoriquement et mathématiquement.

Inconvénients :

- Temps d'entraînement très long avec des jeux de données volumineux.
- Moins efficace sur les jeux de données contenant des bruits.

2.3.1.2 K-plus proche voisins 'k-NN'

L'algorithme des k plus proches voisins (K-Nearest Neighbor 'k-NN') fait partie des méthodes de classification les plus couramment utilisées. Il permet, tout comme les SVM, de traiter des nuages de points non linéairement séparables. Cette approche a l'avantage d'être à la fois simple et efficace. L'algorithme k-NN figure parmi les algorithmes simples d'apprentissage artificiel. Dans un contexte de classification d'une nouvelle observation x , l'idée fondatrice est de faire voter les plus proches voisins de cette observation. La classe de x est déterminée en fonction de la classe majoritaire parmi les k plus proches voisins de l'observation x . La méthode k-NN est donc une méthode basée sur le voisinage, non-paramétrique ; ceci signifiant que l'algorithme permet de faire une classification sans faire d'hypothèse sur la fonction qui relie la variable dépendante aux variables indépendantes [50].

La méthode k-NN : La méthode des k plus proches voisins est une extension de l'idée précédente, qui est largement utilisée. La plus proche observation n'est plus la seule

observation utilisée pour la classification. Nous utilisons désormais les k plus proches observations. Ainsi la décision est en faveur de la classe majoritairement représentée par les k voisins. Soit k_A le nombre d'observations issues du groupe des plus proches voisins appartenant à la classe A.

La méthode 1-NN : c'est une méthode non paramétrique où une nouvelle observation est classée dans la classe d'appartenance de l'observation de l'échantillon d'apprentissage qui lui est la plus proche, vis-à-vis des co-variables utilisées. La détermination de leur similarité est basée sur des mesures de distance. Les distances les plus utilisées sont :

- **La distance Euclidienne :** calcule la racine carrée de la somme du carré différence entre les coordonnées de deux points.
- **La distance de Manhattan :** calcule la somme des valeurs absolues des différences entre les coordonnées de deux points.
- **La distance de Hamming :** est une distance entre deux points donnée simulée par une différence maximale entre leurs coordonnées sur une cote.

Avantages :

- Simple à mettre en œuvre et facile.
- Efficace pour des classes réparties de manière irrégulière.
- La classification est rapide par rapport aux autres approches.
- Reste toujours efficace même si les données ont été incomplètes.

Inconvénients :

- Nécessite une capacité de stockage importante.
- De nombreuses données de références sont nécessaires pour classifier les entrées.

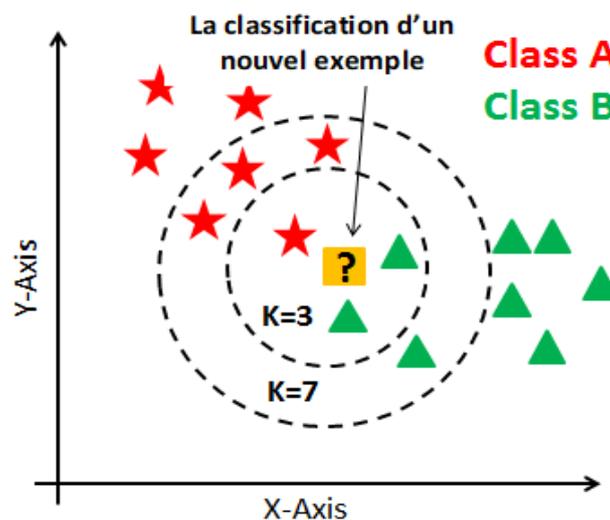


Figure 18 : Schéma représentant le principe intuitif d'un K-NN [51].

2.3.1.3 Réseaux bayésiens naïf

La classification bayésienne naïve (Naive Bayes NB') est un type de classification bayésienne probabiliste simple basée sur le théorème de Bayes avec une forte autonomie (dite naïve) d'hypothèses, issue des travaux du Thomas Bayes (1702-1761) [55]. Il implémente un classificateur bayésien naïf appartenant à la famille des classificateurs linéaires dont le rôle est de classer en groupes (classes) des échantillons aux propriétés similaires, mesurés sur des observations [56]. Le classificateur naïf de Bayes est basé sur le théorème de Bayes, Il s'agit d'un résultat de base en théorie des probabilités, Cette dernière est un classique de la théorie des probabilités.

Ce théorème est basé sur des probabilités conditionnelles. Une probabilité conditionnelle peut être définie intuitivement par : "Quelle est la probabilité qu'un événement se produise sachant qu'un autre événement s'est déjà produit".

Principe mathématique :

Les classificateurs de Naïve Bayes sont une famille d'algorithmes reposant sur le principe commun selon lequel la valeur d'une fonctionnalité spécifique est indépendante de la valeur de toute autre fonctionnalité. Ils nous permettent de prédire la probabilité qu'un événement se produise en fonction de conditions que nous connaissons pour les événements en question. Le principe mathématique peut être formulé [55] :

$$P(A \setminus B) = \frac{P(B \setminus A) P(A)}{P(B)}$$

Où :

A et B : Événements et $P(B) > 0$.

$P(A|B)$: Probabilité que l'événement A se produise sachant que l'événement B s'est produit.

$P(B|A)$: Probabilité que l'événement B se produise sachant que l'événement A s'est produit.

$P(A)$ et $P(B)$: Probabilités des événements A et B indépendamment les uns des autres.

Avantages :

- Robuste aux caractéristiques non pertinentes.
- Relativement simple à appréhender et à implémenter.
- Facile et pratique même avec un petit jeu de données
- La classification est rapide par rapport aux autres MLs.

Inconvénients :

- Théoriquement chaque fonctionnalité est indépendante, ce qui n'est pas toujours le cas dans les exemples réels.

- En général, ce type de classificateur permet de faire le même travail de classification que les autres algorithmes qui existent déjà, mais ces performances sont limitées

2.3.1.4 Réseaux de neurones

Un réseau de neurones est constitué d'un grand nombre de neurones organisées en couches. Un neurone [57] est un objet qui reçoit un signal et lors d'une phase, dite d'activation. Les réseaux de neurones, fortement inspirés du système nerveux biologique, sont composés d'éléments simples appelés neurones, répartis sous forme de couches [79]. Ces éléments sont connectés entre eux par des liaisons affectées de poids (les synapses). L'apprentissage de ces poids est effectué de sorte qu'à chaque vecteur d'attributs sélectionné $Y_{i, j}$ présenté à l'entrée du réseau, corresponde en sortie la classe C_j à laquelle le prototype ω_i , j correspondant appartient.

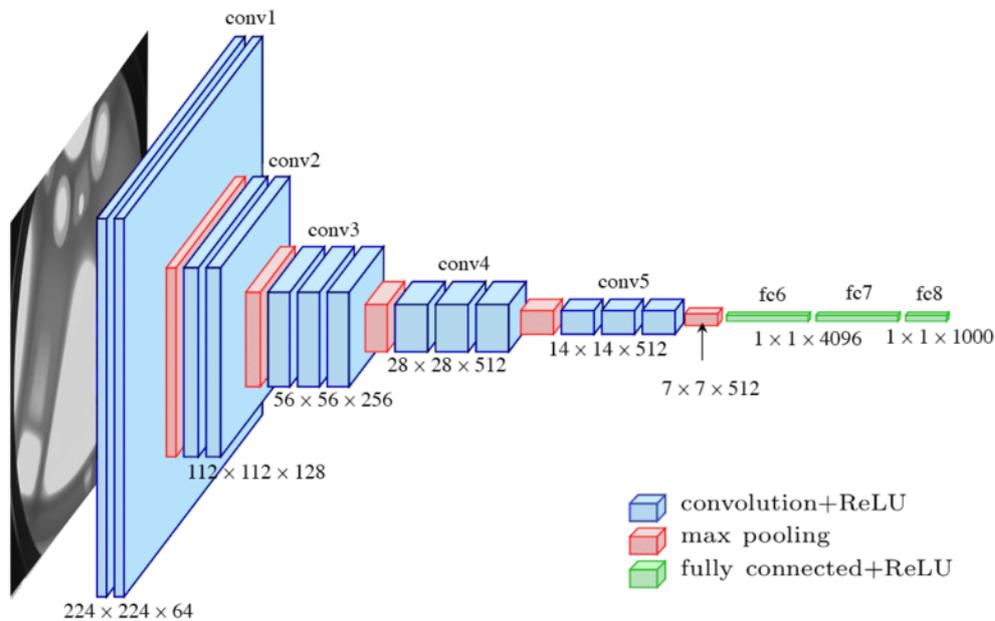


Figure 19 : La structure d'un réseau de neurone CNN [58].

La Figure 13 illustre la structure d'un neurone artificiel. Chaque neurone reçoit des valeurs provenant des neurones en amont à travers ses connexions synaptiques et traite ces valeurs via une fonction de combinaison. Le résultat de la combinaison est alors transformé par la fonction d'activation afin de produire sa sortie. elle permet de seuiliser le résultat pour obtenir une sortie binaire. **Il existe plusieurs types d'architectures pour les réseaux de neurones :**

- Perceptrons Multicouches (MLP) [59], les plus anciens et les plus simples.
- Réseaux Neurones Convolutionnel (CNN) [60], adaptés au traitement d'images.
- Réseaux Neurones Récurrents, adaptés au traitement de texte ou chronologiques [78]

L'interconnexion des neurones, entre eux, forme une couche. Chaque réseau est constitué de 3 couches, d'entrée, de sortie et des couches intermédiaires entre l'entrée et la sortie du réseau. Ces dernières sont appelées aussi couches cachées. Le rôle de ces couches est de transformer les données d'entrée pour en extraire ses caractéristiques qui seront par la suite plus facilement classifiées par la couche de sortie. Dans ces réseaux, l'information se propage de couche en couche, parfois à l'intérieur même d'une couche, par l'intermédiaire de connexions pondérées.

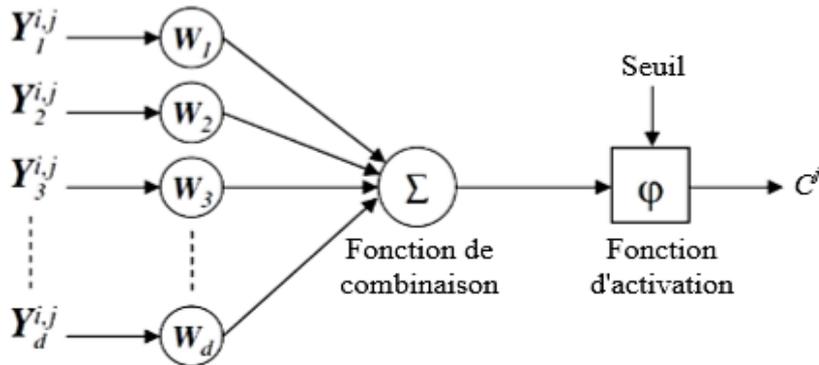


Figure 20 : La structure d'un réseau de neurone [61].

Un réseau de neurones opère en deux phases consécutives : une phase de **conception** et une phase **d'utilisation**. La première phase consiste à choisir l'architecture du réseau et de ses paramètres à savoir le nombre de couches cachées et le nombre de neurones dans chacune de ces couches. Une fois ces choix fixés, il est possible d'entraîner le réseau. Au cours de cette phase (apprentissage), les poids des connexions du réseau et le seuil de chaque neurone seront modifiés pour pouvoir s'adapter aux différentes conditions d'entrée. Une fois l'apprentissage du réseau terminé, le réseau passe à la phase d'utilisation pour réaliser le travail pour lequel il a été conçu. **On peut citer quelques réseaux neurones qui sont largement utilisée :**

- ❖ Réseaux de neurones stochastiques
- ❖ Réseaux de neurones convolutifs

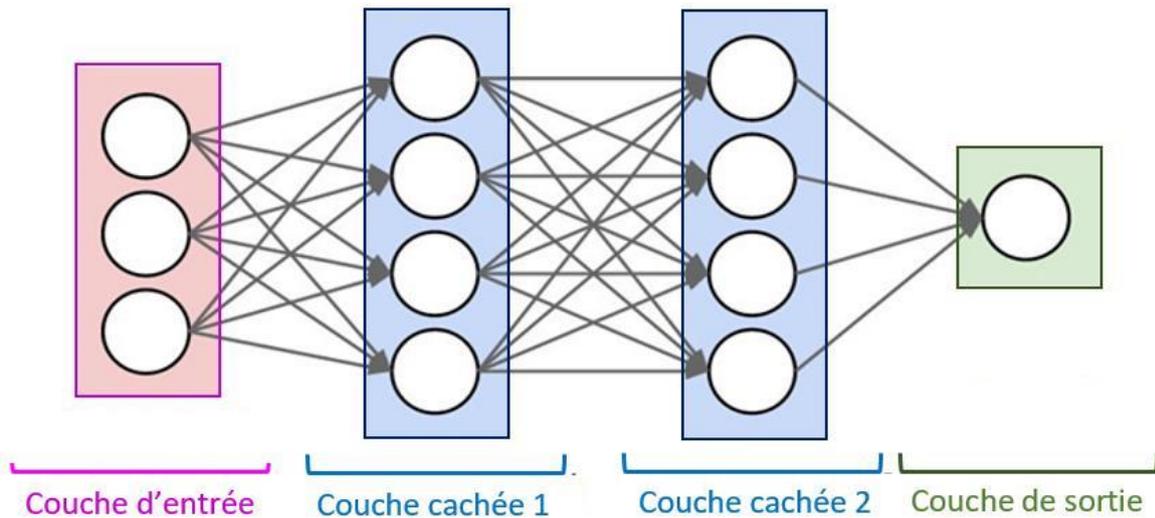


Figure 21 : La structure d'un réseau de neurone multicouches MLP [62].

Le réseau de neurones le plus populaire est le perceptron multicouches (Multi-Layer Perceptron (MLP)). Sa structure est simple, il est constitué de plusieurs couches de neurones (une couche d'entrée, une couche de sortie et une ou plusieurs couches cachées) entièrement connectées entre elles (Figure 15) [80]. Il existe de nombreux autres types de réseaux de neurones, comme par exemple les réseaux à fonction radiale (Radial Basic Functions (RBF)), qui ont pour architecture celle des MLP et pour fonction d'activation une gaussienne, ou encore les réseaux de Hopfield qui permettent un retour en arrière (feed-back).

Les réseaux de neurones récurrents (Recurrent Neural Networks, RNN) [63] sont un type spécifique du réseau neurone interconnecté dans lequel la sortie de l'étape précédente est envoyée en entrée à l'étape actuelle. Dans les réseaux neuronaux traditionnels, toutes les entrées et sorties sont indépendantes les unes des autres, mais dans des cas comme lorsqu'il est nécessaire de prédire le mot suivant d'une phrase, les mots précédents sont requis et doivent donc être mémorisés (voir la figure 16).

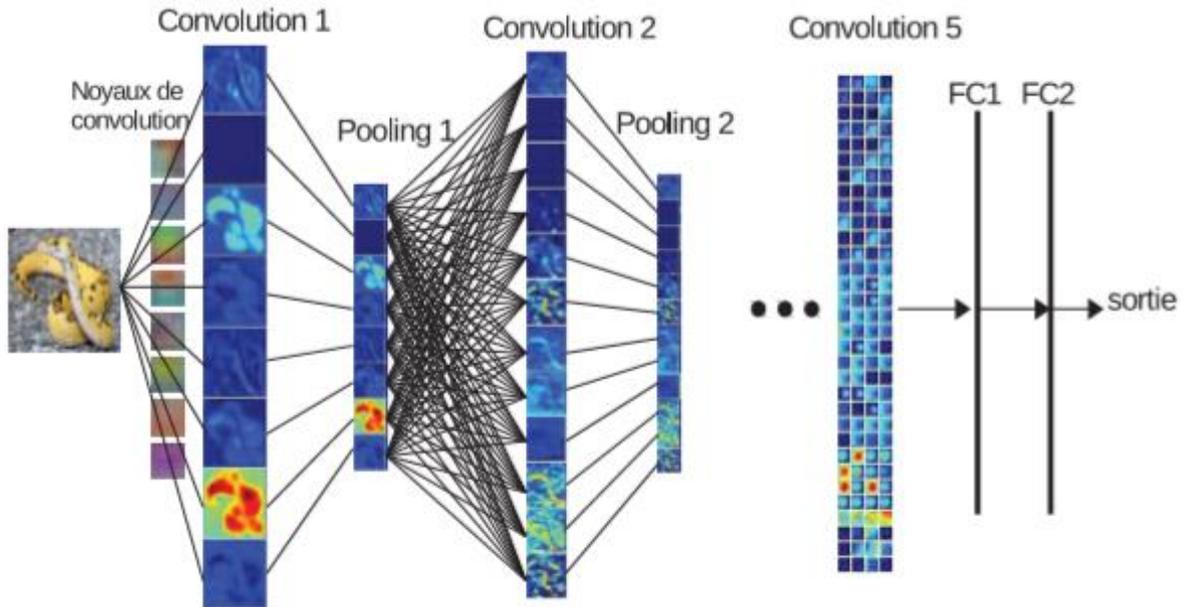


Figure 22 : Exemple d'un réseau de convolutions avec deux couches.

C'est ainsi que RNN a vu le jour, qui a résolu ce problème avec l'aide d'une couche cachée. La caractéristique primordiale de RNN est l'état caché, qui mémorise de certaines informations spécifiques sur une séquence [63].

Avantages :

- Meilleure prédiction des séries temporelles.
- Possède une mémoire à court terme pour mémoriser également les entrées précédentes.
- Voisinage effectif des pixels.

Inconvénients :

- La souffrance des problèmes de fuite et d'explosion.
- L'apprentissage est très complexe.
- Temps de calcul long dans la construction et la représentation des grandes séquences.

2.3.1.5 Arbre de décision

L'arbre de décision (Decision Tree, DT), est une classe des techniques qui se basent sur l'apprentissage automatique supervisé. L'arbre de décision est très similaire au raisonnement humain. Cela nous éclaire également avec beaucoup d'informations sur les données et, surtout, l'arbre de décision est facile à interpréter. Le DT est une technique de classification simple et largement utilisée. Le DT applique une idée rigoureuse pour résoudre le problème de classification en posant une série de questions soigneusement conçues sur les attributs du test.

Chaque fois qu'il reçoit une réponse, une autre question est posée jusqu'à ce qu'une conclusion soit atteinte concernant l'étiquette de la classe du dossier [65].

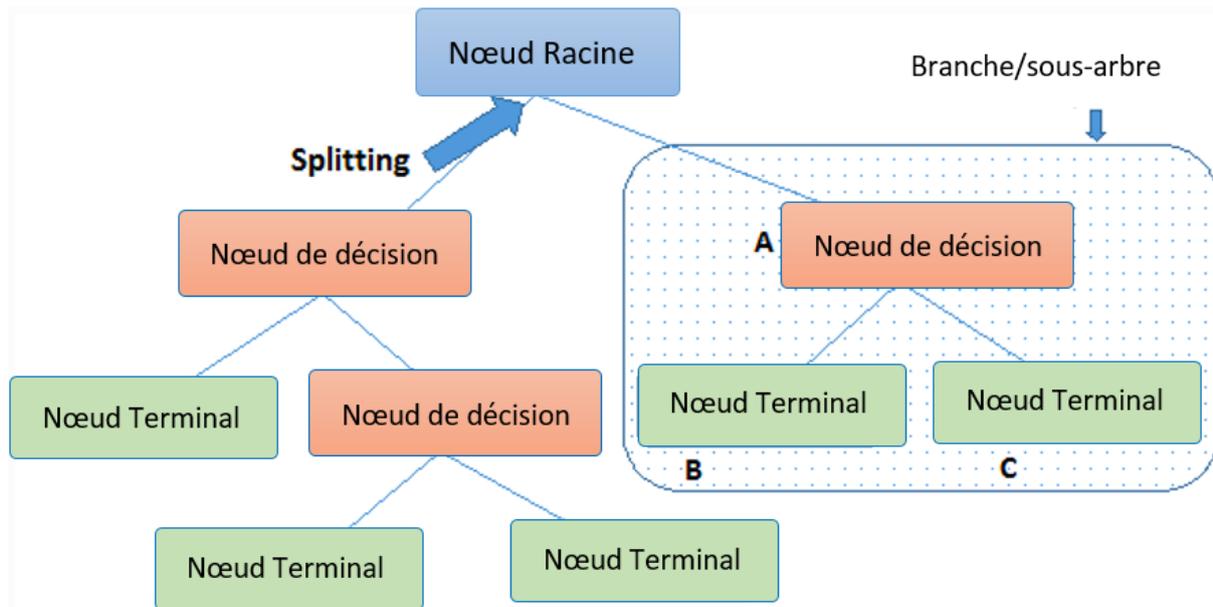


Figure 23: La structure générale d'un arbre de décision DT [64].

Un arbre de décision permet de créer un modèle d'apprentissage qui peut être utilisé pour prédire la classe ou la valeur de la variable cible en apprenant des règles de décision simples déduites de données antérieures (données d'entraînement). Dans les arbres de décisions, pour prédire une classe, la méthode commence de parcourir l'arbre de la racine. Ensuite elle compare les valeurs de l'attribut racine avec l'attribut de l'enregistrement. Sur la base de la comparaison, nous suivons la branche correspondant à cette valeur et passons au nœud suivant. Il existe plusieurs algorithmes utilisés dans les arbres de décision, l'algorithme ID3 [66] , [67], C4.5 le successeur d'ID3 (Itérative Dichotomiser 3), CART (Classification and Regression Tree), CHAID (Détection automatique des interactions du chi-square Effectue des divisions à plusieurs niveaux lors du calcul des arbres de classification) et MARS (Multivariate Adaptive Regression Splines). [68]

Avantages :

- Principe simple et peu sensible aux variables non discriminantes.
- Facile de les associer à d'autres d'outils de décisions.
- Aucun modèle et aucun présupposé à satisfaire.
- Règles simples et facilement interprétables.
- Traitement des valeurs manquantes.

Inconvénients :

- Souffre du problème sur apprentissage.
- N'est pas robuste au nombre de classes.
- Nécessite d'un grand nombre d'individus.

2.3.1.6 Apprentissage en profond

Le Deep Learning (DL), ou apprentissage profond, est la principale technologie d'IA. Le DL repose sur la technologie des réseaux de neurones. Ils ont constitué de plusieurs couches artificielles interconnectés [73-75, 76]. Plus le nombre des neurones est élevé plus le réseau est profond, d'où vient le nom "Deep". Le terme "Deep Learning" a été introduit pour la première fois par Dechter (1986) [73], et aux réseaux neuronaux par Aizenberg et al (2000) [74].

Le Deep Learning est un nouveau domaine de recherche du ML, qui a été introduit dans le but de concevoir une machine procède de la même façon de celle d'un être humain. Ils peuvent apprendre plusieurs niveaux de représentation dans le but de modéliser des relations complexes entre les données. Il s'agit d'algorithmes inspirés par la structure et le fonctionnement du cerveau. L'apprentissage profond repose sur l'utilisation d'une nouvelle technologie des réseaux de neurones artificiels pour exploiter et gérer des quantités massives de données tout en ajoutant des couches au réseau. Ces multiples couches permettent au réseau de neurones DL d'extraire des caractéristiques complexes à partir des données brutes à l'aide de multiples transformations linéaires et non linéaires à travers les couches multiples [75].

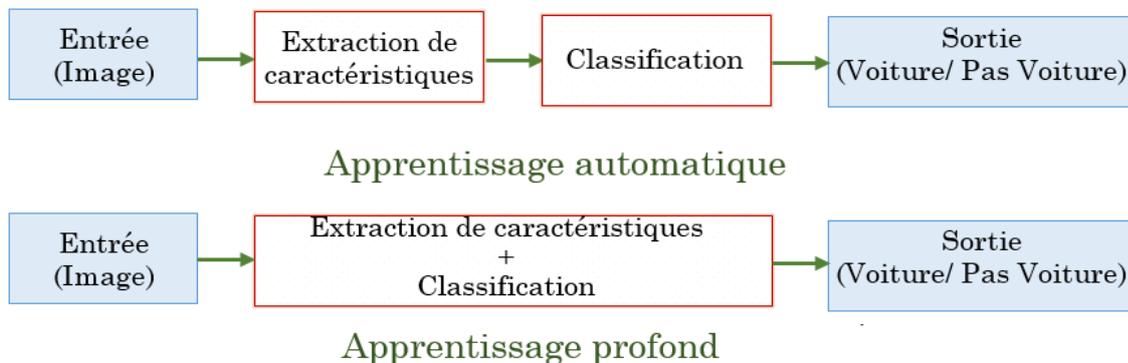


Figure 24 : Schéma général du ML comparé à celui du DL.

Avantages :

- Caractéristiques automatiquement déduites et optimisées pour le résultat souhaité.
- Robustesse aux variations apprises automatiquement.
- Même niveau de réutilisabilité que les réseaux de neurones.
- Calculs massivement parallèles grâce à l'utilisation de GPU évolutifs pour traiter de gros volumes de données

Inconvénients :

- Technique difficile et ambiguë avec un calcul intensif.
- Certains problèmes se prêtent mieux au DL que d'autres applications.

- Modèles plus simples peuvent être suffisants pour certains domaines de problèmes.
- Les modèles sont très difficiles à expliquer par rapport aux autres algorithmes ML.

2.3.2 Classification non supervisée

Dans le cas d'un apprentissage non supervisé, l'apprentissage automatique se déroule de manière autonome. Des données sont alors communiquées à la machine sans lui fournir les exemples de résultats attendus en sortie. Il appartient à l'algorithme de déterminer les critères les plus pertinents pour classer les données. Le but de l'apprentissage non supervisé est de modéliser la structure de base ou la distribution sous-jacente dans les données afin d'en savoir plus sur les données. Celles-ci sont appelées apprentissage non supervisé car, contrairement à l'apprentissage supervisé ci-dessus, il n'y a ni bonne réponse ni superviseur.

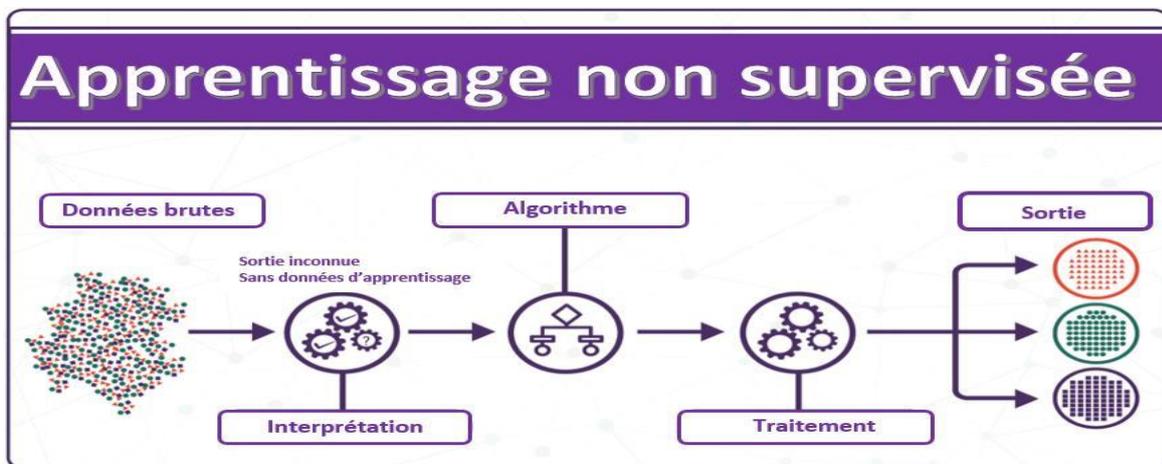


Figure 25 : Schéma général d'un système d'apprentissage non supervisé [44].

Cette méthode de classification est aussi appelée "classification automatique", (clustering en anglais) ou encore "regroupement". Aucune information a priori sur la texture n'est connue. On cherche alors à regrouper les différents exemples de textures à traiter en fonction de la valeur de leurs descripteurs de manière à créer des classes homogènes. On suppose qu'on dispose d'un ensemble d'objets que l'on note par $X = \{x_1, x_2, \dots, x_N\}$ caractérisé par un ensemble de descripteurs "D", l'objectif du regroupement est de trouver les groupes auxquels appartiennent chaque objet "x" on note par $C = \{c_1, c_2, \dots, c_N\}$.

Ce qui revient à déterminer une fonction notée "Ys" qui associe à chaque élément de X un ou plusieurs éléments de C. Il faut pouvoir affecter une nouvelle observation à une classe. Les observations disponibles ne sont pas initialement identifiées comme appartenant à telle ou telle population. L'absence d'étiquette de classe est un lourd handicap qui n'est que très partiellement surmontable. Ce procédé nécessite généralement de fixer au préalable le nombre de classes désirées, que ce soit de manière empirique ou automatique [45].

Parmi les méthodes non-supervisées les plus utilisées, citons deux types d'approches : les centres mobiles (k-means) et la classification hiérarchique. Les problèmes d'apprentissage non supervisés peuvent être classés en problèmes de Clustering et d'association [47] :

- ❖ **Clustering** : permet de découvrir les regroupements inhérents aux données, tels que le regroupement des clients en fonction du comportement d'achat.
- ❖ **Association** : permet de découvrir des règles décrivant une grande partie de données, telles que les acheteurs de X ont également tendance à acheter Y.

L'évaluation non-supervisée des résultats de segmentation, consiste à exécuter des mesures d'homogénéité des régions ou de contrastes entre régions, Stella et Shi dans [49], ont classifié ces critères en deux grandes catégories : Les critères de Contraste et les critères d'équation au modèle. Les premiers s'intéressent à la variabilité inter-régions, alors que les seconds s'intéressent à l'uniformité en intensité ou en couleur à l'intérieur des régions. [49]

2.3.2.1 K-means

Il existe différentes approches de clustering [69]. L'approche K-means est parmi les approches les plus utilisées. K-means est l'une des méthodes les plus connues pour la classification non supervisée créée par J. Mac Queen en 1967. Elle nécessite comme unique paramètre, le nombre de classes.

L'algorithme k-means permet de répartir un ensemble de données (individus ou pixels de l'image) en k classes homogènes. La valeur de k n'est pas connue, elle est choisie selon une connaissance préalable (combien de régions sont souhaitées) ou de manière aléatoire. Chaque groupe est représenté par son centre de gravité qui correspond à la moyenne des éléments (pixels) du groupe. Cette méthode de classification est basée sur le mouvement de ces centres, qui se déplacent vers une position de stabilité après chaque itération.

Il est très utilisé de par sa simplicité de mise en œuvre et sa faible complexité de calcul. L'approche débute par l'initialisation préalable des centres de chaque cluster puis les affine au fil d'itérations [70]. **Le déroulement de l'algorithme est le suivant :**

- ❖ Association de chaque vecteur au centre du cluster le plus proche.
- ❖ Calcul des nouvelles coordonnées du centre par rapport aux vecteurs qui y sont associés.

Les itérations se poursuivent jusqu'à ce qu'à atteindre une condition terminale, qui peut être un nombre fixe d'itérations, un nombre minimal de vecteurs assignés à un autre cluster ou encore des conditions sur l'erreur quadratique entre un centre et les vecteurs qui y sont associées (oscillations, arrivée à un minimum, ...).

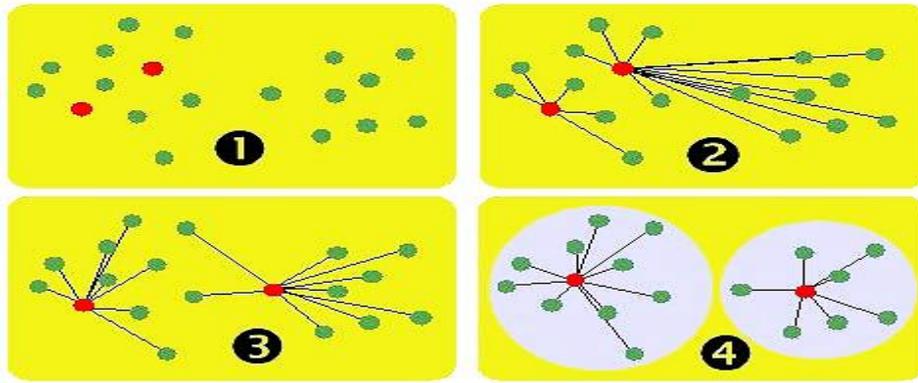


Figure 26: Exemple de classification avec la méthode de k-means.

À la fin de la classification par les k-means, on obtient des classes strictes, c'est-à-dire que chaque individu n'appartient qu'à une seule classe. Les individus situés en périphérie d'une classe, près de la frontière d'une autre classe, posent le problème d'appartenance stricte à une classe unique [71].

Les principales étapes de l'algorithme k-means sont :

1. Choix aléatoire de la position initiale des K clusters.
2. Réaffecter les objets à un cluster suivant un critère de minimisation des distances
3. Une fois tous les objets placés, recalculer les K centroïdes.
4. Répéter les étapes 2 et 3 jusqu'à ce que plus aucune réaffectation ne soit faite [72].

2.4 Conclusion

Dans ce chapitre, nous avons présenté et détaillé le principe général de l'apprentissage automatique, les deux approches essentielles de l'apprentissage automatique, supervisé et non supervisé. Il existe deux types de problèmes de classification qui s'adaptent mieux à l'apprentissage automatique supervisé, tels que la classification des textures.

Cependant, il existe un autre type de problème qui nécessite un apprentissage non supervisé. Ensuite nous avons présenté les approches de classifications les plus connues et les plus utilisées dans le domaine de reconnaissance d'objets et plus particulièrement dans le domaine de la reconnaissance faciale et la reconnaissance des expressions faciales.

Chapitre 3

‘Approche Proposée pour l'Extraction des Caractéristiques’

3.1 Introduction

Le but global de la classification de la texture est d'atteindre deux objectifs principaux, le premier est de capturer les informations de texture les plus représentatives et spécifiques d'une classe de texture afin de pouvoir distinguer entre les différentes classes de texture. Le deuxième est d'assurer une faible complexité de calcul, pour permettre à la tâche de classification des images.

3.2 Environnements de travail

3.2.1 Environnement PyCharm

PyCharm est un outil développé par la société JetBrains, spécialiste au développement des IDEs pour différents langages de programmation. Pour ce qui est de PyCharm, il existe deux versions de PyCharm :

- **La version communautaire** : elle est gratuite et déjà très riche. Tous les outils nécessaires à l'exécution de ce tutoriel sont présents dans cette version. Elle est non limitée dans le temps, mais, fonctionnellement parlant, il s'agit d'un sous-ensemble des possibilités offertes par la version professionnelle.
- **La version professionnelle** : cette version payante, contient tous les assistants proposés dans la version communautaire mais propose de nombreux autres outils orientés développement Web, calculs scientifiques et machine learning. Il en existe une version d'essai, limité dans le temps. Je ne vous conseille pas de la tester maintenant, car les extensions proposées nous seraient tout simplement inutile pour le moment. Attendez de progresser un peu et quand le besoin s'en fera sentir, vous pourrez tester cette version et profiter pleinement de la période d'essai.



Figure 27 Logo de logiciel PyCharm.

3.2.2 Environnement Pythone

Python est un langage de programmation open source le plus employé par les informaticiens. Ce langage s'est propulsé en tête de la gestion d'infrastructure, d'analyse de données ou dans le domaine du développement de logiciels. En effet, parmi ses qualités, Python permet notamment aux développeurs de se concentrer sur ce qu'ils font plutôt que sur la manière dont ils le font. Il a libéré les développeurs des contraintes de formes qui occupaient leur temps avec les langages plus anciens. Ainsi, développer du code avec Python est plus rapide qu'avec d'autres langages. Il reste aussi accessible pour les débutants, à condition de lui consacrer un peu de temps pour la prise en main. Les développeurs utilisent Python principalement pour :

- **La programmation d'applications.**
- **La création de services web.**
- **La génération de code.**
- **La métaprogrammation.**



Figure 28: Python logo

3.2.3 Environnement Qt designer

Qt designer est une série d'outils intégrés dans Qt Creator permettant de créer graphiquement des interfaces graphiques de l'application.



Figure 29 Qt designer logo.

3.3 Méthode proposée pour la classification des textures

La méthode proposée pour l'extraction des caractéristiques dans ce travail est la méthode LCV (Locale Corrélation Voisin) est une nouvelle méthode (original) basé sur le mixage entre l'algorithme LBP et la corrélation. Figure 30 montre l'architecture d'un système de reconnaissance des textures.

3.3.1 Descripteur LBP

Le descripteur LBP est un processus système pour la reconnaissance des textures, il permet l'extraction des caractéristiques d'une texture. L'un des inconvénients de l'opérateur fondamental de LBP est qu'il est incapable de saisir certaines caractéristiques dominantes. Afin de pouvoir traiter la texture à des échelles complètement différentes, il a donc fallu travailler sur l'opérateur et ainsi émerger l'utilisation de voisinage de différentes tailles. Un voisinage local est défini comme un groupe de points d'échantillonnage de la même dimension répartis sur un cercle et centrés à l'intérieur des pixels. Le descripteur LBP est principalement proposé pour la première fois pour analyser de texture et du traitement de classification des données. La figure suivante exprime l'architecture d'apprentissage LBP :

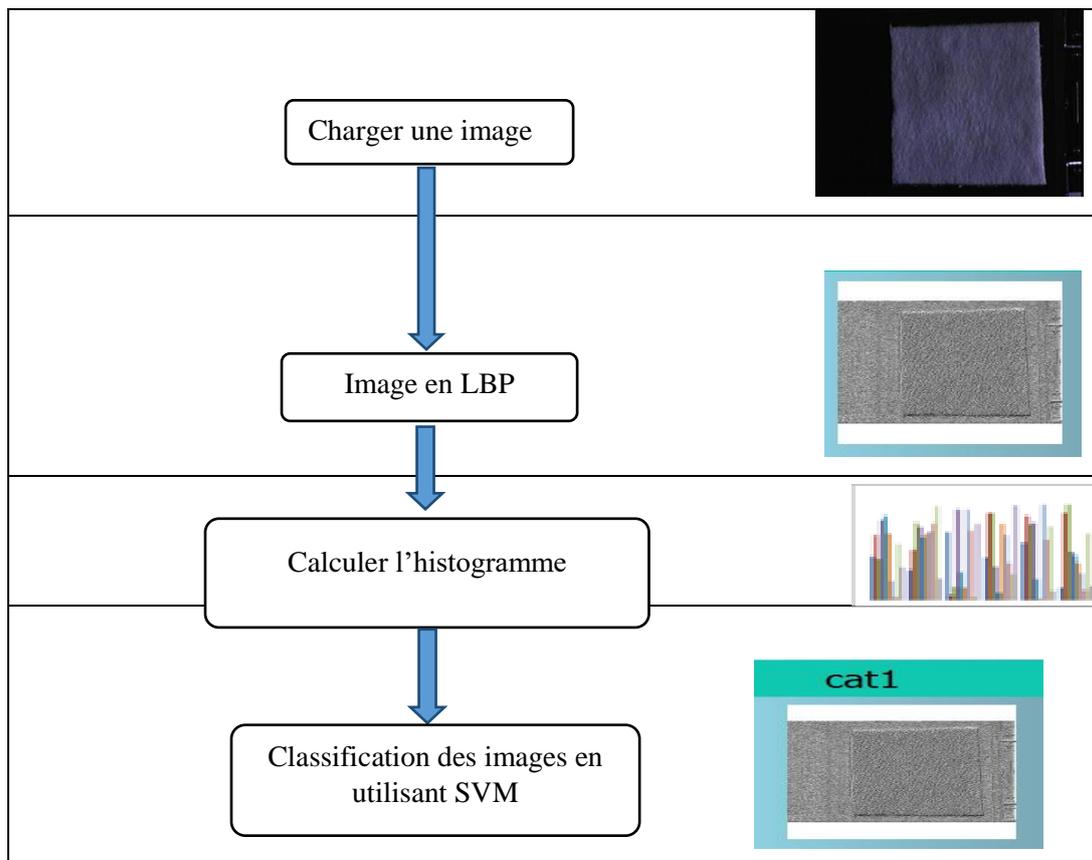


Figure 31 Architecture d'un système de reconnaissance des textures.

L'histogramme est l'un des moyens les plus couramment utilisé pour capturer les informations visuelles d'une image. Il présente de nombreux avantages, tels que l'invariance aux rotations de l'image et la robustesse des translations de l'image autour de l'axe de visualisation. Les résultats de classification qui utilise l'histogramme n'étaient pas très prometteurs, ce qui a conduit à de nombreuses variantes des modèles de partitionnement d'une image.

3.3.2 Descripteur LCV

L'approche proposée pour l'extraction des caractéristiques est d'utiliser le descripteur LCV (Locale Corrélation Voisin). C'est un descripteur basé sur le mixage entre l'algorithme LBP et la corrélation.

Le modèle d'architecture de la méthode LCV utilisé pour l'apprentissage est le suivant :

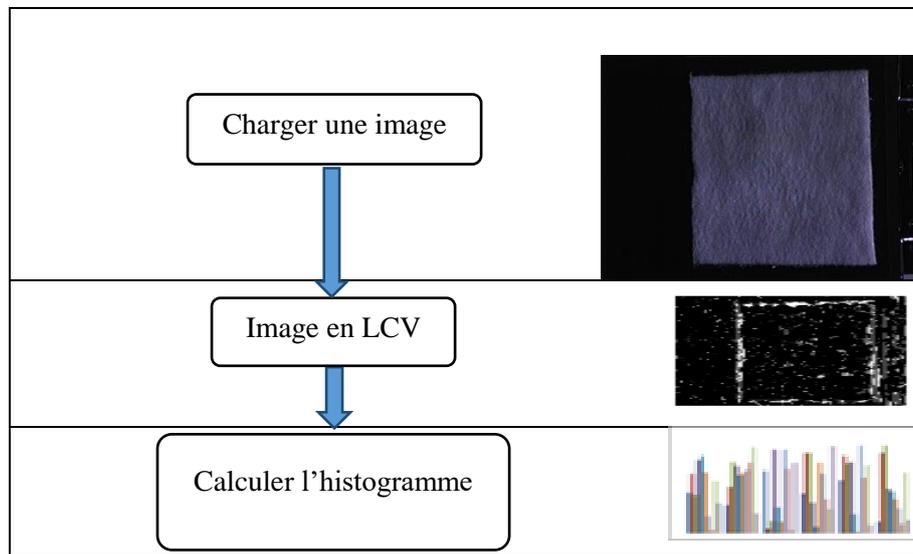


Figure 32 Modèle d'apprentissage par la méthode LCV.

3.3.2.1 Corrélation

La corrélation est une mesure statistique qui exprime la notion de liaison linéaire entre deux variables (ce qui veut dire qu'elles évoluent ensemble à une vitesse constante). C'est un outil courant permettant de décrire des relations simples sans s'occuper de la cause et de l'effet. Toujours la corrélation donné une valeur entre [-1 ,1]

$$r = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^n (X_i - \bar{X})^2 \sum_{i=1}^n (Y_i - \bar{Y})^2}} \quad (3.1)$$

3.3.2.2 Algorithme LCV

Dans la figure suivante nous avons donné un exemple qui représente une structure matricielle d'une photo pour le connaitre par l'ordinateur, elle contient des valeurs entières :

14	200	8	10	16	1
5	12	15	15	20	19
6	255	14	9	2	0
9	100	16	20	17	20
10	19	16	20	11	20
1	0	55	8	100	11

Figure 33 Photo matricielle.

Pour modifier un pixel il faut prendre tous les voisins de ce pixel, dans notre cas le pixel est la cellule sélectionnée en jaune «14 ».

Figure 34 Les voisins de la matrice centrée,

Le calcul de cette modification, commence par la corrélation entre la matrice centrée et les autres matrices voisins, si la valeur absolue de la corrélation est supérieure à une valeur donné « 0.75 » cette valeur va représenter la similarité entre les deux matrices (centré, voisins) :

- En cas supérieur de 0.75 : donné un 1.
- En cas inférieur de 0.75 : donné un 0.

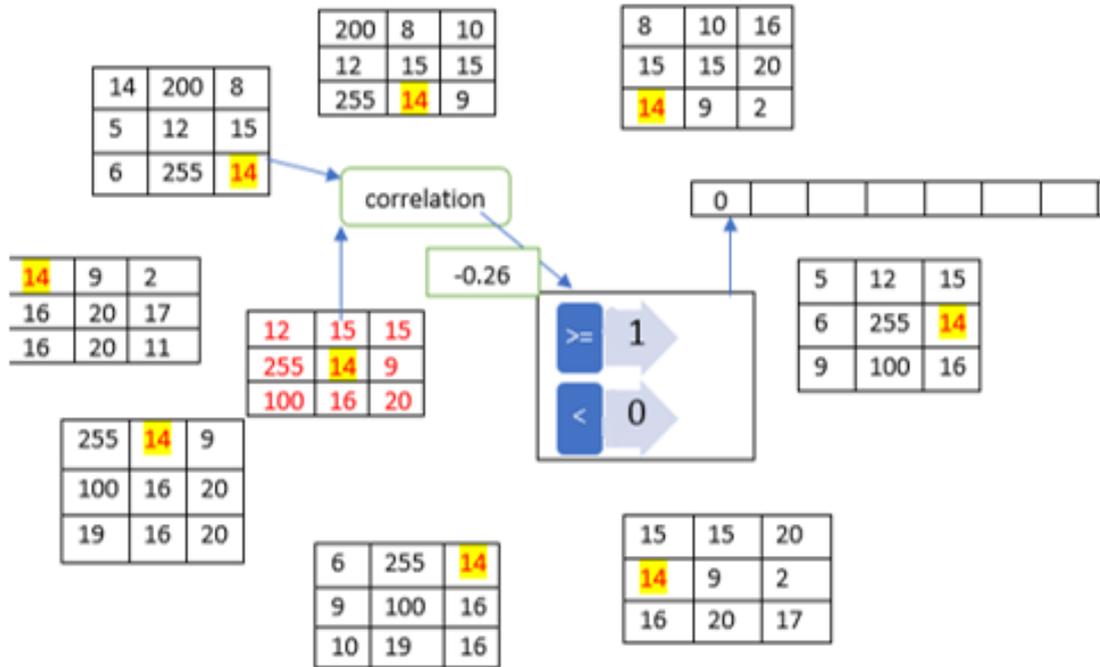


Figure 35 Le calcul d'un bit.

Ces calculs peuvent être appliqués aux autres matrices voisines.

La figure ci-dessus représente la transformation de la matrice globale :

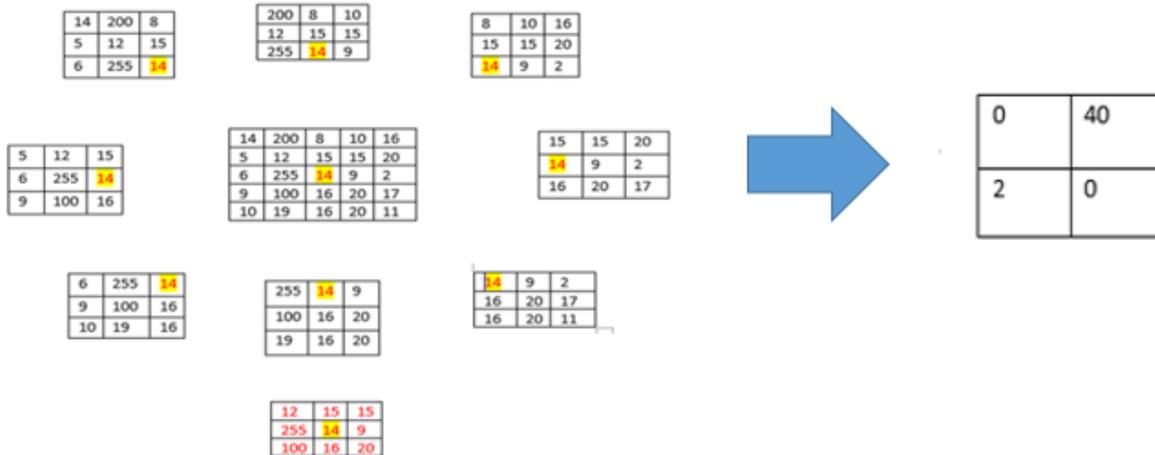


Figure 36 : La transformation finale de l'image.

3.4 Expérience et résultats

3.4.1 Base de données

Dans nos expériences, nous avons utilisé la base de données CUREt. C'est une base de données d'image et de texture qui contient 61 catégories. Pour tester les performances de notre

application de classification de texture, nous avons pris 70 % des images de cette BDD pour l'apprentissage et le reste des images est utilisé pour le test.

3.4.2 Résultats obtenus

Pour évaluer la performance de notre approche, nous avons utilisé l'équation 3.2 pour calculer la précision de classification de texture.

$$\text{Précision} = \frac{T_p}{T_p + F_p} \quad (3.2)$$

TP : True Positive

FP : False Positive

Nous avons réalisé l'application à l'aide du langage Python version '3.9'. Les résultats de classification de texture sont affichés dans la figure 38 qui montre l'utilisation des descripteurs LBP et LCV.

Les résultats de comparaison de notre approche par rapport à l'approche de LBP montrent que les résultats obtenus sont modestes.

Les résultats de classification de textures en utilisant le descripteur LCV permet de donner un résultat de 78.6%, dont le classificateur SVM est utilisé sur la base de données CUREt. Les expériences sur la même BDD qui utilisent le descripteur LBP ont donné une précision de 93.6 %.

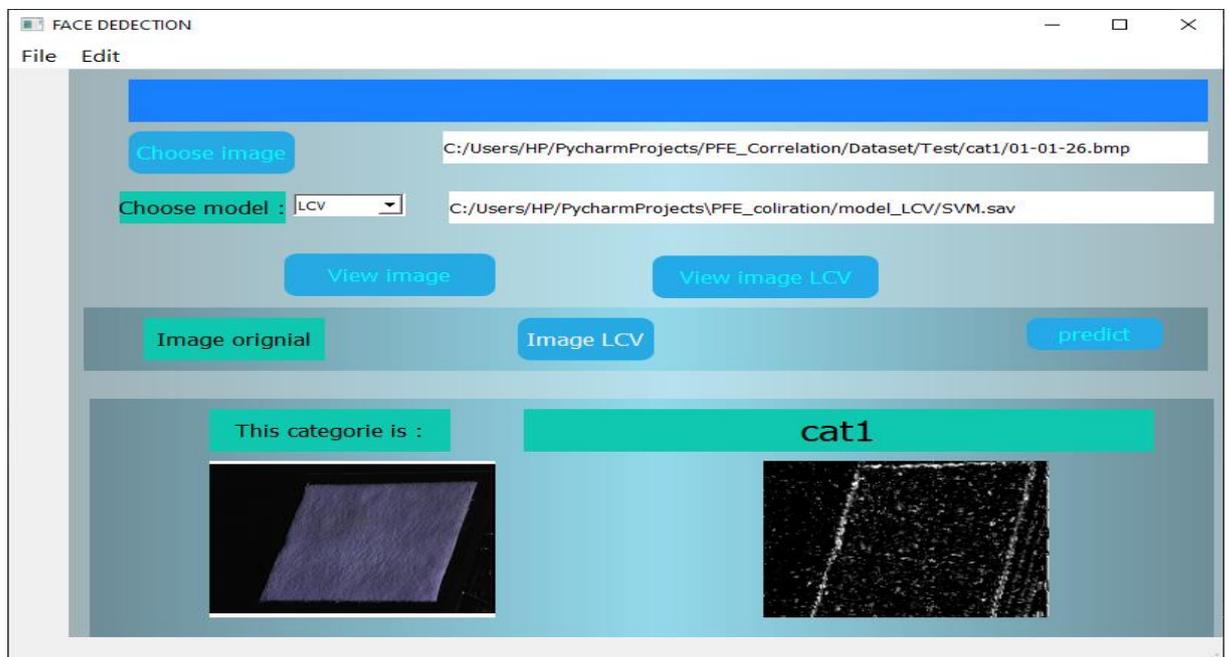


Figure 37 Interface de l'application de reconnaissance des expressions textures LCV

3.5 Conclusion

Nous avons présenté dans ce chapitre en détail le principe général de la nouvelle méthode LCV ainsi que les résultats obtenus des expériences effectuées sur la BBD CURet pour classifier les images de texture.

« ----- »

Conclusion générale

Dans ce mémoire, nous tentons de participer à l'amélioration des performances des systèmes de classification de texture. Nous utilisons le descripteur LCV est utilisé pour la description de l'information pertinentes dans l'image. Ce descripteur est construit en basant sur le descripteur LBP pour construire un vecteur qui représente l'image de texture qui peut améliorer les performances finales de la classification des images. Des expérimentations réalisées sur la BDD CURet avec la méthode SVM démontrent l'efficacité de l'approche proposée. Bien que nos résultats soient modestes.

Pour cela, nous prévoyons d'améliorer notre modèle de classification en exploitant des informations spatiales dans l'image. Nous prévoyons également d'exploiter les algorithmes d'apprentissage en profondeur pour fournir une comparaison complète entre les différents modèles d'entraînement en termes de précision, de temps d'exécution.

- [1] Bolon, P., Chassery, J., Cocquerez, J., Demigny, D., Graffigne, C., Philipp. S. (1995). Analyse d'images: Filtrage et segmentation. MASSON, Enseignement de la physique, 2-225-84923-4.
- [2] Chakib, K. (1999). Généralités sur le traitement d'image. Site www.ImgDist/LM.fr/travs/C1/1999.
- [3] Swain, M., Ballard, D. (1991). International Journal of computer vision, 32(11) :11-32.
- [4] Souvannavong, F., Merialdo, B., & Huet, B. (2005). Region-based video content indexing and retrieval. Fourth International Workshop on Content-Based Multimedia Indexing.
- [5] OJALA, T., PIETIKÄINEN, M., HARWOOD, D. (2002). A comparative study of texture measures with classification based on feature distributions. Pattern Recognition. Vol. 29, Num. 1, 51–59.
- [6] OJALA T., PIETIKÄINEN M., MÄENPÄÄ T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. Pattern Recognition. Vol. 24, Num. 7.
- [7] SOLTANA W. B., HUANG D., ARDABILIAN M., CHEN L., AMAR C. B. (2010). Comparison of 2d/3d features and their adaptive score level fusion for 3d face recognition. 3D Data Processing, Visualization and Transmission (3DPVT), Paris, 17–20.
- [8] JAIN A., NANDAKUMAR K., ROSS A. (2005). Score normalization in multimodal biometric systems. Pattern Recognition. Vol. 38, Num. 12, 2270–2285.
- [9] JAIN A. K., DUIN R. P. W., MAO J. (2000). Statistical pattern recognition : A review. IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. 22, Num. 1, 4–37.
- [10] SOLTANA W. B., HUANG D., ARDABILIAN M., CHEN L., AMAR C. B. (2011). A mixture of gated experts optimized using simulated annealing for 3D face recognition. International Conference on Image Processing (ICIP), Brussels, 11–14.
- [11] RAO, A.R. (1990). A taxonomy for texture description and identification, Springer series in Perception engineering, Springer-Verlag, Heidelberg,
- [12] Julesz, B. (1962). 'Visuel pattern recognition', IEEE Trans. on Information Theory, vol. 8.
- [13] Gagalowicz. A. (1983). Vers un modèle de textures, Thèse d'État, Université Pierre et Marie Curie, Paris VI.
- [14] Volet, P. (1987). 'Analyse et synthèse d'images de textures structurées, Thèse de l'Ecole Polytechnique de Lausanne,
- [15] Coggins, J. M. (1983). A Framework for Texture Analysis Based on Spatial Filtering. Thèse de doctorat, Michigan State University.
- [16] Sklansky, J. (1978). Image segmentation and feature extraction. IEEE Transactions on Systems, Man, and Cybernetics, 8:237–247.
- [17] Haralick, R. M. (1979). Statistical and structural approaches to textures. Proceedings of the IEEE, 67(5):786–804.
- [18] Unser, M. (1984). Description statistique de la texture. Application à l'inspection automatique. Thèse de doctorat, EPFL, Lausanne.

- [19] Armi, L., Shervan F.(2019). Texture image analysis and texture classification methods - a review International Online Journal of Image Processing and Pattern Recognition Vol. 2, No.1, pp. 1-29.
- [20] Rahim, M.A., Azam, M.S., Hossain, N., and Islam, M.R. (2013). Face recognition using local binary patterns (LBP).Global Journal of Computer Science and Technology, 1-12.
- [21] Nagy, A.M., Ahmed, A., and Zayed, H. (2014). Particle filter based on joint color texture histogram for object tracking, Paper presented at the Proceeding of the First International Conference on Image Processing, Applications and Systems Conference (IPAS), pp. 1-6.
- [22] Raheja, J. L., Kumar, S., and Chaudhary, A. (2013). Fabric defect detection based on GLCM and Gabor filter: A comparison, Optik-International Journal for Light and Electron Optics, 124(23).
- [23] Patterns, M. L. (2017). Breast density classification using multiresolution local quinary patterns in Mammograms, Paper presented at the Proceeding of the 21st Annual Conference in Medical Image Understanding and Analysis, pp. 365-376, Edinburgh, UK.
- [24] Duque, J. C., Patino, J. E., Ruiz, L. A., and Pardo-Pascual, J. E. (2015). Measuring intra-urban poverty using land Cover and texture metrics derived from remote sensing data, Landscape and Urban Planning, 135, 11-21.
- [25] Wood, E. M., Pidgeon, A. M., Radeloff, V. C., &Keuler, N. S. (2012). Image texture as a remotely sensed measure of vegetation structure. Remote Sensing of Environment, 121, 516-526.
- [26] Gool, L. J. V., Dewaele, P. et Oosterlinck, A. (1985). Texture analysis anno 1983. Computer Vision, Graphics, and Image Processing, 29(3):336–357.
- [27] Unser, M. (1995). Texture classification and segmentation using waveletframes. IEEE Transactions on Image Processing, 4:1549–1560.
- [28] Brodatz, P. (1966). Textures : A Photographic Album for Artists andDesigner. Dover Pub.
- [29] AHONEN T., MATAS J., PIETIKÄINEN M. : Rotation invariant image description with local binary pattern histogram Fourier features. Image Analysis (2009), 61–70.
- [30] GUOYING Z., AHONEN T., MATAS J., PIETIKÄINEN M. (2012). Rotation-invariant image and video description with local binary pattern features. IEEE Transactions on Image Processing. Vol. 21, Num. 4, 1465–1477.
- [31] Tuceryan, M. et Jain, A. K. (1998). Texture Analysis, chapitre 2.1, pages 207–248. The Handbook of Pattern Recognition and Computer Vision.
- [32] Harrington, P. (2012). Machine Learning in Action. Manning Publications Co, Green-wich, CT, USA, ISBN 1617290181, 9781617290183.
- [33] Ojala, T. Pietikäinen, M. Mäenpää, T. (2002).Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Trans. Pattern Anal. Mach. Intell. 24 (7), 971–987.
- [34] BELHADJ, M.(2019). Etude et simulation d'un système de reconnaissance des expressions faciale', Université Mohamed Khider de Biskra Faculté des Sciences et de la Technologie Département de Génie Electrique.

- [35] MANGAT, A., K. Ranade, Sukhjeet Kaur. (2016). Facial Expression Recognition Based on Local Binary Patterns. *International Journal of Advanced Research in Computer Science*, , vol. 7.
- [36] HUANG, Di, SHAN, Caifeng, ARDABILIAN, Mohsen, A. (2011). Local binary patterns and its application to facial image analysis: a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 41, no 6, p. 765-781.
- [37] CARCAGNÌ, Pierluigi, DEL COCO, Marco, LEO, Marco. (2015). Facial expression recognition and histograms of oriented gradients: a comprehensive study. *Springer Plus*, vol. 4, no 1, p. 645.
- [38] ORRITE, Carlos, GAÑÁN, Andrés, et ROGEZ, Grégory. (2009). Hog-based decision tree for facial expression classification. In: *Iberian Conference on Pattern Recognition and Image Analysis*. Springer, Berlin, Heidelberg, p. 176-183.
- [39] https://fr.wikipedia.org/wiki/Histogramme_de_gradient_orient%C3%A9.
- [40] MIGNIOT, C. (2012). Segmentation de personnes dans les images et les vidéos. Thèse de doctorat. Grenoble.
- [41] Ben Soltana, W., Porebski, A., Vandebroucke, N., Ahmad, A., et Hamad, D. (2014). Contribution des descripteurs de texture LBP à la classification d'images de dentelles, *Colloque Compression et Représentation des Signaux Audiovisuels*, Reims Image, Laboratoire d'Informatique Signal et Image de la Côte d'Opale Maison de la Recherche Blaise Pascal 50, rue Ferdinand Buisson, France.
- [42] Kevin, P., Murphy. (2012). *Machine Learning: A Probabilistic Perspective* (Adaptive Computation and Machine Learning series). The MIT Press.
- [43] Pat Langley and Herbert, Simon, A. (1995). Applications of machine learning and rule induction. *Communications ACM*, 38(11):54-64.
- [44] <https://datafloq.com/read/machine-learning-explained-understanding-learning/4478>.
- [45] Park, H., and Park, K. R. (2007). Iris recognition based on score level fusion by using SVM," *Pattern Recognition Letters*, vol. 28, pp. 2019-2028.
- [46] Cortes, C. and Vapnik, V. (1995). Support vector networks", *Machine Learning*, pp. 273–297.
- [47] Marshland, S. (2015). *Machine learning: An algorithm perspective*. Boca Raton, FL: CRC Press.
- [48] Alpayđın, E. (2014). *Introduction to machine learning*. Cambridge, MA: MIT Press.
- [49] Mitchell, T. M. (1999). Machine learning and data mining. *Communications ACM*, 42(11):30-36.
- [50] Goldstein, M. "k-Nearest Neighbor Classification," *Information Theory*, *IEEE Transactions*
- [51] <https://www.datacamp.com/community/tutorials/k-nearest-neighbor-classification-scikit-learn> on, vol. 18, pp. 627-630, (1972).
- [52] Fan, R. E., Chen, P. H., Lin, C. J. (2005). Working set selection using second-order information for training support vector machines. *Journal of Machine Learning Research*, 6, 1889-1918.

- [53] [https://en.wikipedia.org/wiki/Supportvector machine#/media/File:KernelMachine](https://en.wikipedia.org/wiki/Supportvector_machine#/media/File:KernelMachine)
- [54] Dahmane, M., Meunier, J. (2011). Emotion recognition using dynamic grid-based HoG features. In: Face and Gesture 2011 (p. 884-888).
- [55] Berry, D. (1996). Statistics—A Bayesian Perspective, Duxbury Press.
- [56] Lewis, D. D. (1998). Naive at forty: the independence assumption in information retrieval, in: C. Nédellec, C. Rouveirol (Eds.), Machine Learning: ECML-98: 10th European Conference on Machine Learning, Chemnitz, Germany, April 21–23, Springer, Berlin/Heidelberg, 1998.
- [57] Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, pages 65–386.
- [58] Karen Simonyan, Andrew Zisserman (2015) Very Deep Convolutional Networks for Large-Scale Image Recognition, *Computer Vision and Pattern Recognition*.
- [59] Bishop (1995): *Neural networks for pattern recognition*, Oxford University Press.
- [60] Le, C. Y., Jackel L., Boser B., Denker J., Graf H., Guyon I., Henderson D., Howard R., and Hubbard W. (1998) Handwritten digit recognition: Applications of neural networks chips and automatic learning. *Proceedings of the IEEE*, 86(11):2278–2324.
- [61] https://commons.wikimedia.org/wiki/File:ArtificialNeuronModel_francais.png?uselang=fr.
- [62] <http://blog.christianperone.com>.
- [63] Hochreiter S. and Schmidhuber J. (1997) Long short-term memory. *Neural Computation*, 9(8):1735–1780.
- [64] <https://www.kdnuggets.com/2020/01/decision-tree-algorithm-explained.html>
- [65] Gershman A, Meisels A, Lüke KH, Rokach L, Schclar A, Sturm A. (2010). A Decision Tree Based Recommender System. In *IICS*, pp. 170-179.
- [66] Mingu Kang, Sujun K. Gonugondla, Sungmin Lim, Naresh R. (2018). Shanbhag A 19.4-nJ/Decision, 364-K Decisions/s, In-Memory Random Forest Multi-Class Inference Accelerator, *IEEE Journal of Solid-State Circuits*. Volume: 53, Issue: 7.
- [67] <https://www.kaggle.com/c/visum-facial-expression-analysis>.
- [68] Priyama A, Abhijeeta RG, Ratheeb A, Srivastava S. (2013). Comparative analysis of decision tree classification algorithms. *International Journal of Current Engineering and Technology*. 3(2):334,7.
- [69] Xu, R. and Wunsch-II, D. (2005). Survey of clustering algorithms", *IEEE Transactions on Neural Networks*, 16, 3, p. 645--678.
- [70] Hartigan, J. and Wang, M.(1979). A k-means clustering algorithm, *Applied Statistics*, 28, 100–108
- [71] Ludovic Paulhac. Outils et méthodes d'analyse d'images 3D textures : Application à la segmentation des images échographiques. Thèse doctorat de l'université François – Rabelais
- [72] Frédéric, D., Brahim, E. Etude de méthodes de Clustering pour la segmentation d'«images en couleurs», *Faculté Polytechnique de Mons*. Les pages 1,2.

Bibliographies

- [73] Dechter and J. Pearl, (1986). ‘The cycle-cutset method for improving search performance in AI applications. University of California, Computer Science Department.
- [74] Aizenberg, I. Aizenberg, N. N. and Vandewalle, J. P. (2013). Multi-Valued and Universal Binary Neurons: Theory, Learning and Applications. Springer Science & Business Media.
- [75] Schmidhuber, J. (2015). Deep learning. Scholarpedia, vol. 10, no. 11, p. 32832.
- [76] Shaveta Dargan, Munish Kumar, Maruthi Rohit Ayyagari, Gulshan Kumar. (2020) A Survey of Deep Learning and Its Applications: A New Paradigm to Machine Learning. Archives of Computational Methods in Engineering. Vol 27, pages 1071–1092.
- [77] Gao, J. and Tan, P.N. (2006). Converting output scores from outlier detection algorithms into probability estimates. In Proceedings of the Sixth International Conference on Data Mining, ICDM '06, pages 212–221, Washington, DC, USA. IEEE Computer Society.
- [78] <https://heartbeat.fritz.ai/introduction-to-basic-object-detection-algorithms-b77295a95a63>
- [79] Rojas, R. (1996). Neural Networks: A Systematic Introduction: Springer Berlin Heidelberg.
- [80] Gevers, T. van de Weijer, J. and Stokman, H. (2006). Color image processing: methods and applications: color feature detection: an overview, chapter 9, pages 203–226. CRC press.